

General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Modeling and Analysis of PINHOLE OCCULTER EXPERIMENT

Contract No. NAS8-36101

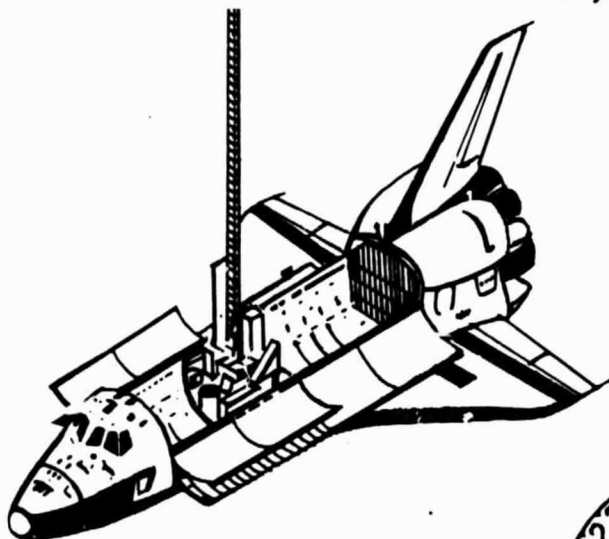
Initial Study Phase
FINAL REPORT

(NASA-CR-175978) MODELING AND ANALYSIS OF
PINHOLE OCCULTER EXPERIMENT: INITIAL STUDY
PHASE Final Report (Honeywell, Inc.) 64 p
HC A04/MF A01 CSCI 14B

NE5-30287

Unclas
21593

G1/35



Honeywell

SPACE & STRATEGIC
AVIONICS DIVISION
CLEARWATER, FLORIDA



MODELING AND ANALYSIS
of
PINHOLE OCCULTER EXPERIMENT

Contract No. NAS8-36101

Initial Study Phase

FINAL REPORT

Prepared by: R. J. VanderVoort

Date: May 20, 1985

Table of Contents

| | Page |
|---------------------------------------|------|
| 1.0 INTRODUCTION | 1 |
| 2.0 CONCLUSIONS | 2 |
| 3.0 RECOMMENDATIONS | 3 |
| 4.0 DISCUSSION | 4 |
| 4.1 Math Models | 4 |
| 4.1.1 Structure Math Model | 4 |
| 4.1.2 Sensor Math Model | 16 |
| 4.1.3 Actuator Math Model | 21 |
| 4.1.4 Controller Math Model | 24 |
| 4.2 Full State Controller | 31 |
| 4.2.1 Introduction | 31 |
| 4.2.2 Salient Features | 31 |
| 4.2.3 Design Methodology | 31 |
| 4.3 Performance Analysis | 38 |
| 5.0 COMPUTER LISTINGS | 43 |

List of Abbreviations

DOF - Degree of Freedom

IMU - Inertial Measurement Unit

INT - TREETOPS INTERACTIVE file

MSFC - Marshall Space Flight Center

P/OF - Pinhole/Occulter Facility

VRCS - Vernier Reaction Control System

wrt - with respect to

1.0 INTRODUCTION

The objective of this contract was twofold:

- 1.) To demonstrate the feasibility of using a generic simulation, TREETOPS, to simulate the Pinhole/Occulter Facility (P/OF).
- 2.) To determine the pointing performance of the P/OF using the baseline control system.

In order to accomplish these objectives two tasks were defined:

TASK 1.1 Modeling and TREETOPS setup

This task included modeling the structure as a three body problem including the flexibility of the P/OF 32 meter boom. Modeling of the sensors, actuators and control algorithm was also required.

TASK 1.2 TREETOPS Setup, Simulation and Analysis

The TREETOPS simulation was defined based on the models of Task 1.1. Transient responses were obtained for comparison with previous analysis results.

During the execution of these tasks the direction of the contract was changed such that Honeywell would design and supply a control algorithm for use in TREETOPS and the comparison with previous results would be deleted.

The detailed math models for the structure, sensors and actuators is presented in Section 4.1. The control algorithm and corresponding design procedure is presented in Section 4.2. The closed loop pointing performance using this controller is presented in Section 4.3. Computer listings of the simulation are presented in Section 5 for the sake of completeness.

2.0 CONCLUSIONS

Two major conclusions were reached as a result of the study contract.

1.) The TREETOPS simulation is an excellent tool for this type application. Its generic nature gives us the flexibility to change models easily and the linearization option provides a natural link to control design tools.

2.) Steady state P/OF pointing to within 1 arcsecond appears feasible.

It should be noted that the emphasis of this contract was on the feasibility of simulation via TREETOPS and on control algorithm design as opposed to emphasizing closed loop performance assessment. The transient pointing errors during orbiter maneuver periods were large (47 arcsec) but when the maneuvers were complete the steady state errors were small (less than 1 arcsec). Our conclusion is that steady state errors will depend largely on sensor errors and noise rather than loop dynamics.

The next phase of this contract will model sensor and actuator noise and thus give a much better prediction of the steady state pointing error.

3.0 RECOMMENDATIONS

The objective of this contract was primarily to develop a TREETOPS based simulation of the P/OF. The simulation proved to be very useful and our recommendation is to use the simulation for further studies oriented toward understanding the performance limitations of the P/OF. Specifically, we recommend three areas of study that will provide significant near term benefits.

1. **IMPLEMENTATION** - This task would put the pointing control system in a form suitable for digital computer implementation and would try to minimize the computer resources required for implementation.

2. **STEADY-STATE PERFORMANCE** - The addition of sensor and actuator error sources to the simulation would allow sensitivity studies for determining the relationship between pointing accuracy and control system characteristics. These relationships would form the basis for predicting pointing accuracy and the associated error budget allocations.

3. **MOUNTING BASE** - The current simulation utilizes the shuttle orbiter as a mounting base. The viability of the P/OF concept would be enhanced if we showed acceptable performance for a variety of bases such as space station, space lab or a free-flier base.

4.0 DISCUSSION

The purpose of this section is to present the results of this study as they relate to the statement of work. Section 4.1 presents the detailed math models developed according to Task 1.1 of the contract. Section 4.2 presents the pointing control system design which was an add on task. Finally, Section 4.3 presents the closed loop performance responses obtained from the simulation according to Task 1.2.

4.1 MATH MODELS

The simulation math model can be divided into four main parts: structure, sensors, actuators and controller.

4.1.1 Structure Math Model

The structure being modeled consists of the Space Shuttle orbiter, Instrument Pointing System (IPS) and Pinhole Occulter facility as illustrated in Figure 1. We are treating the structure as three bodies connected in a chain configuration. The simulation is 3 dimensional (as opposed to planar) and has 9 degrees of freedom in addition to the flexibility of the boom. The orbiter and IPS are both treated as rigid bodies while the boom is a flexible body. The orbiter has 6 degrees of freedom (3 rotation, 3 translation) wrt the inertial frame and the IPS has 3 rotational degrees of freedom wrt the orbiter.

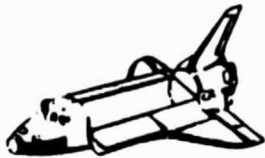
The IPS is modeled as a single 3 DOF hinge which means that the individual gimbals are not modeled. That part of the IPS outboard of the gimbals is combined with the P/OF sensors and the boom deployment canister to make up Body No. 2.

The third body is the 32 meter flexible boom with the P/OF mask assembly mounted at the outboard end. It is connected to the canister with a zero DOF hinge representing a cantilever beam. The flexibility of the boom is represented using the cantilever modes obtained from the finite element model supplied by MSFC. The three bodies and three hinges are illustrated in Figures 2 through 4 and the corresponding TREETOPS input data is presented in Tables 1 through 6.

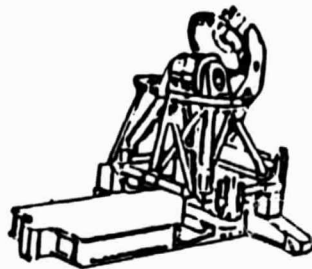
AN ILLUSTRATION OF THE P/OF STRUCTURE

FIGURE 1

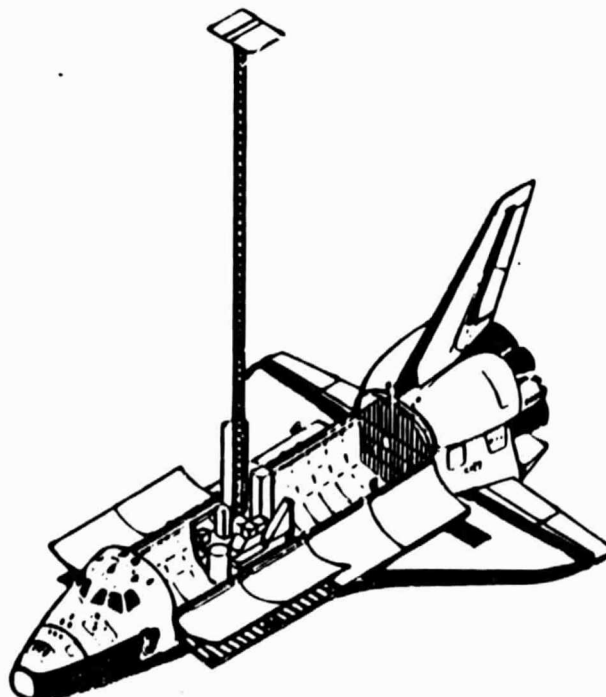
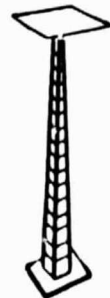
SPACE SHUTTLE



**INSTRUMENT
POINTING SYSTEM
(IPS)**



**PINHOLE OCCULTER
FACILITY**



BODY #1
SPACE SHUTTLE ORBITER

FIGURE 2

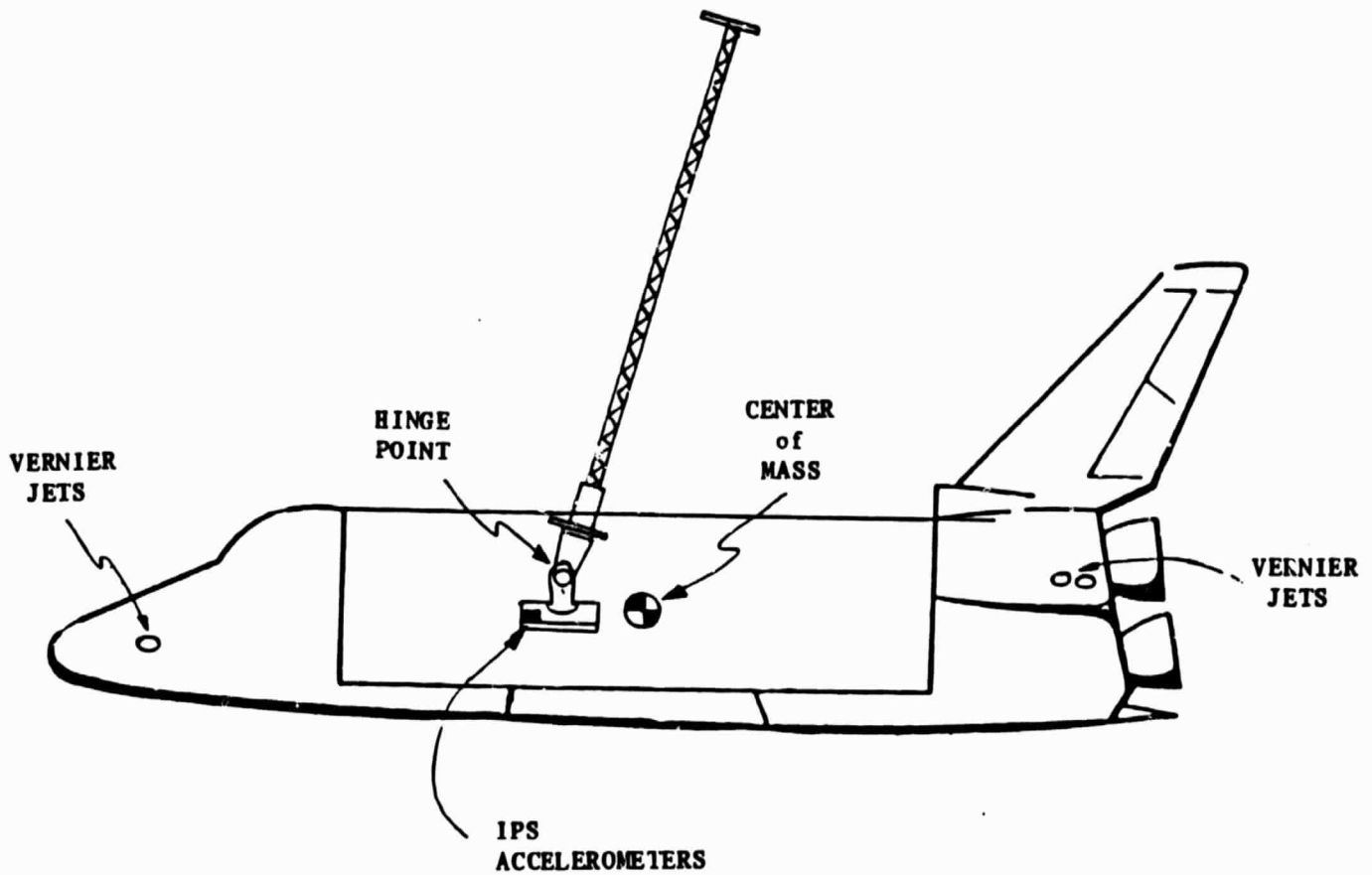
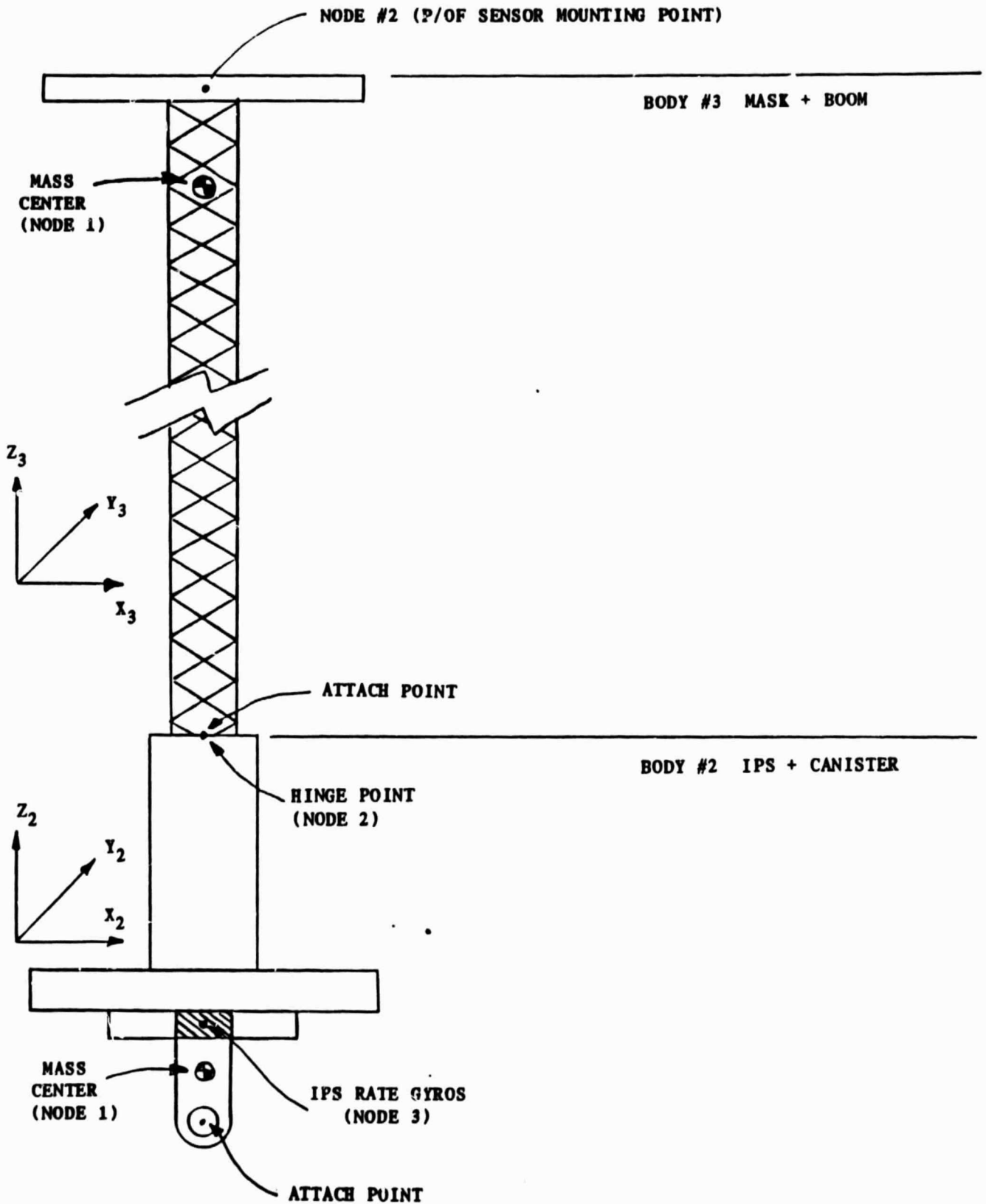
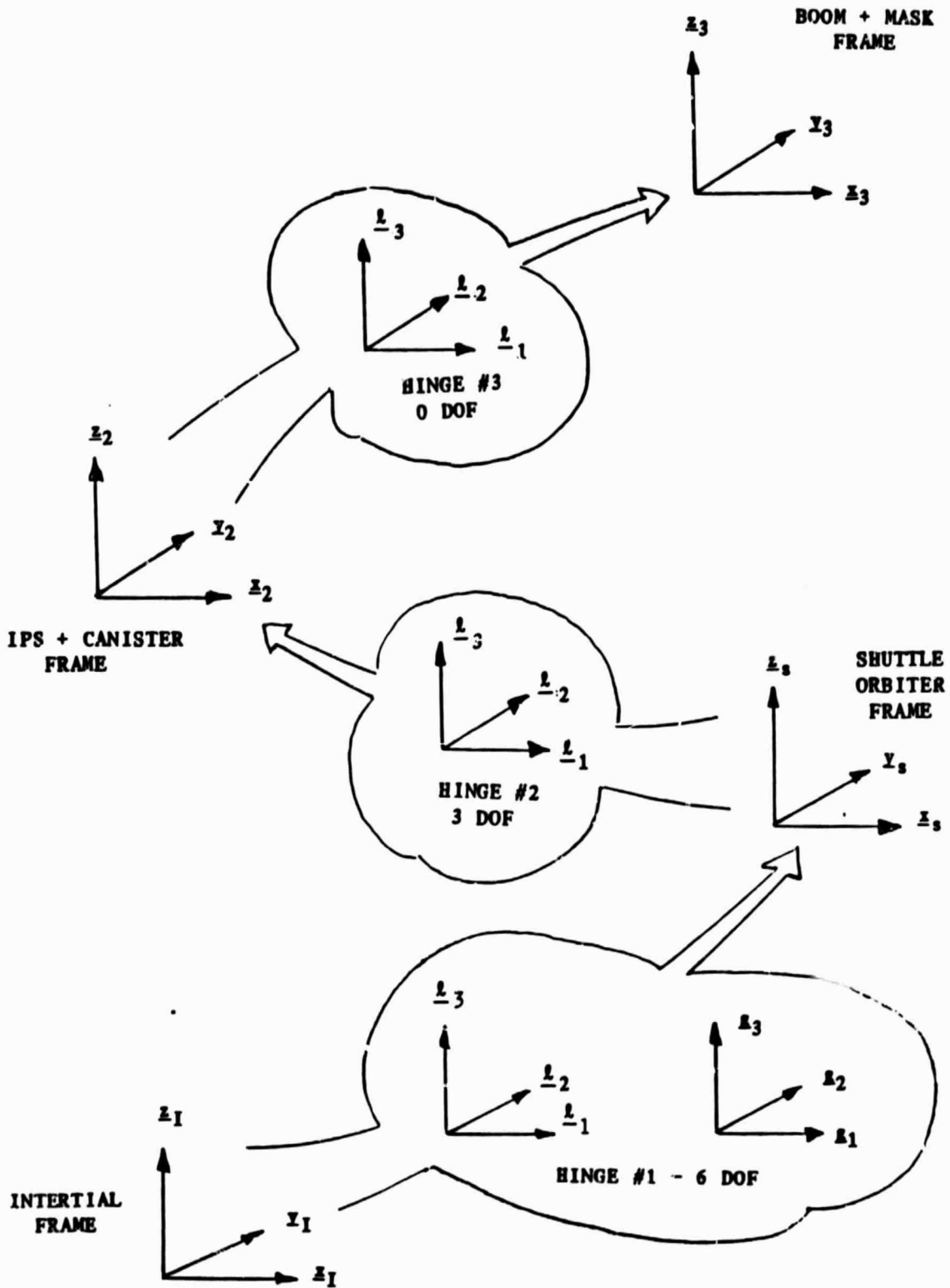


FIGURE 3



P/OF CCORDINATE FRAME AND HINGE ILLUSTRATION

FIGURE 4



BODY #1
SPACE SHUTTLE ORBITER

TABLE 1
(REVISION 1)

- 1.) ID# = 1
- 2.) TYPE = RIGID
- 3.) MASS = 95395 Kg (210,311.5 LBS)
- 4.) INERTIA = (Kg -m²)
- $I_{xx} = 1.261 \times 10^6$ (930,016 slug-ft²)
- $I_{yy} = 9.691 \times 10^6$ (7,148,059 slug-ft²)
- $I_{zz} = 10.150 \times 10^6$ (7,485,903 slug-ft²)
- $I_{xy} = 1337$ (986 slug-ft²)
- $I_{xz} = 3.125 \times 10^5$ (230,475 slug-ft²)
- $I_{yz} = -676.6$ (-499 slug-ft²)
- 5.) ATTACH POINT COORDINATES (meters)
- 27.736, -.0025, 9.469 (1092.0 in, -.1 in, 372.3 in)
- 6.) NODE #1 = MASS CENTER (meters)
- 27.736, -.0025, 9.469 (1092.0, -.1, 372.8 in)

TABLE 1
(CONTINUED)

7.) NODE #2 = HINGE POINT (meters)

25.527, 0, 10.566 (1005, 0, 416 in)

NODE #3 = IPS ACCELEROMETERS (meters)

25.527, 0, 10.26 (1005, 0, 404 in)

NODE #4,5,6,7,8,9 = VERNIER JETS (meters)

#4 F5R 8.24, 1.52, 8.89 (324.4, 59.7, 350.1 in)

#5 F5L 8.24, -1.52, 8.89 (324.4, -59.7, 350.1 in)

#6 R5R 39.75, 3.81, 11.66 (1565., 149.9, 459. in)

#7 L5L 39.75, -3.81, 11.66 (1565., -149.9, 459. in)

#8 R5D 39.75, 3.0, 11.57 (1565., 118., 455.4 in)

#9 L5D 39.75 -3.0, 11.57 (1565., -118., 455.4 in)

BODY #2

IPS + CANISTER

TABLE 2

1.) ID# = 2

2.) TYPE = RIGID

3.) MASS = 1875 Kg

4.) MOMENT OF INERTIA (Kg -m²)

$$I_{xx} = 12691 \quad (9361 \text{ slug-ft}^2)$$

$$I_{yy} = 12354 \quad (9112 \text{ slug-ft}^2)$$

$$I_{zz} = 1903 \quad (1404 \text{ slug-ft}^2)$$

$$I_{xy} = 0$$

$$I_{xz} = 0$$

$$I_{yz} = 0$$

5.) ATTACH POINT COORD. (meters)

0,0,0

6.) NODE #1 = MASS CENTER (meters)

0,0, 1.8 (5.9 ft)

7.) NODE #2 = HINGE POINT (meters)

0,0, 4.3198

NODE #3 = RATE GYRO MOUNTING POINT (meters)

(0,0,1)

BODY #3
BOOM + MASK
TABLE 3

- 1.) ID# = 3
- 2.) TYPE = RIGID
- 3.) MASS = 69.87 Kg
- 4.) MOMENT OF INERTIA ($\text{Kg} \cdot \text{m}^2$)
 - $I_{xx} = 65391$
 - $I_{yy} = 65416$
 - $I_{zz} = 81$

 - $I_{xy} = 0$
 - $I_{xz} = 0$
 - $I_{yz} = 0$
- 5.) ATTACH POINT COORD. (meters)
 - 0,0,0
- 6.) NODE #1 = MASS CENTER (meters)
 - 0,0, 29.92
- 7.) NODE #2 = P/OF SENSOR MOUNTING POINT (meters)
 - (0,0, 32.)

HINGE #1

TABLE 4

| | | |
|------|-------------------------|-------------|
| 1.) | ID# | = 1 |
| 2.) | INBOARD BODY ID# | = 0 |
| 3.) | OUTBOARD BODY ID# | = 1 |
| 4.) | HINGE POINT NODE # | = 1 |
| 5.) | NR | = 3 |
| 6.) | ROTATION OPTION | = GIMBALLED |
| 7.) | $\underline{l}_{1L(j)}$ | = 1,0,0 |
| 8.) | \underline{l}_{1j} | = 1,0,0 |
| 9.) | $\underline{l}_{3L(j)}$ | = 0,0,1 |
| 10.) | \underline{l}_{3j} | = 0,0,1 |
| 11.) | K_{θ} | = 0,0,0 |
| 12.) | B_{θ} | = 0,0,0 |
| 13.) | $\theta(t_0)$ | = 0,0,0 |
| 14.) | θ_{NULL} | = 0,0,0 |
| 15.) | NT | = 3 |
| 16.) | \underline{g}_1^j | = 1,0,0 |
| | \underline{g}_2^j | = 0,1,0 |
| | \underline{g}_3^j | = 0,0,1 |
| 17.) | K_y | = 0,0,0 |
| 18.) | B_y | = 0,0,0 |
| 19.) | $y(t_0)$ | = 0,0,0 |
| 20.) | y_{NULL} | = 0,0,0 |

HINGE #2
IPS GIMBAL
TABLE 5

- 1.) ID# = 2
- 2.) INBOARD BODY ID# = 1
- 3.) OUTBOARD BODY ID# = 2
- 4.) HINGE POINT NODE # = 2
- 5.) NR = 3
- 6.) (ROTATION OPTION - NOT REQUIRED)
- 7.) $\underline{l}_{1L(j)}$ = 1,0,0
- 8.) \underline{l}_{1j} = 1,0,0
- 9.) $\underline{l}_{3L(j)}$ = 0,0,1
- 10.) \underline{l}_{3j} = 0,0,1
- 11.) K_{θ} = 0,0,0
- 12.) B_{θ} = 0,0,0
- 13.) $\theta(t_0)$ = 0,0,0
- 14.) θ_{NULL} = 0,0,0
- 15.) NT = 0

HINGE #3

CANISTER TO BOOM INTERFACE

TABLE 6

- 1.) ID# = 3
- 2.) INBOARD BODY ID# = 2
- 3.) OUTBOARD BODY ID# = 3
- 4.) HINGE POINT NODE # = 2
- 5.) NR = 0
- 6.) (ROTATION OPTION - NOT REQUIRED)
- 7.) $\underline{l}_{1L(j)} = 1,0,0$
- 8.) $\underline{l}_{1j} = 1,0,0$
- 9.) $\underline{l}_{3L(j)} = 0,0,1$
- 10.) $\underline{l}_{3j} = 0,0,1$

4.1.2 Sensor Math Models

The P/OF simulation uses standard rate gyros mounted on the outboard portion of the IPS and accelerometers on the base of the IPS. In addition, we have added a special sensor that measures line of sight error, mask tilt and tip deflection of the P/OF boom.

The two accelerometers are mounted at Node 3 of Body 1. Their input axes are aligned along the orbiter X and Y axes respectively. Node 3 of Body 1 corresponds to a location on the base of the IPS. It should be noted that the accelerometers were to be used for cancelling the disturbance effect of base motion but the technique which was based on a rigid boom did not work well for a flexible boom. The accelerometers remain in the simulation but are not used as part of the control system.

The simulation has three standard rate gyros mounted at Node 3 of Body 2 which is the instrument mounting plate on the outboard end of the IPS. Their input axes are aligned along the X, Y and Z axes of the IPS respectively. The X and Y gyros are used as feedback sensors for the roll and pitch axes respectively and the Z gyro is unused.

The final sensor is a line-of-sight (LOS) sensor developed specifically for the P/OF simulation. This sensor has seven outputs defined as follows:

- 1 β_1)
- 2 β_2) tilt angles (rad) of mask plane wrt detector plane
- 3 β_3)

- 4 x_1) relative displacement (meters) of mask wrt detector due
- 5 x_2) to elastic deformation

- 6 δ_1) line of sight error (radians)
- 7 δ_2)

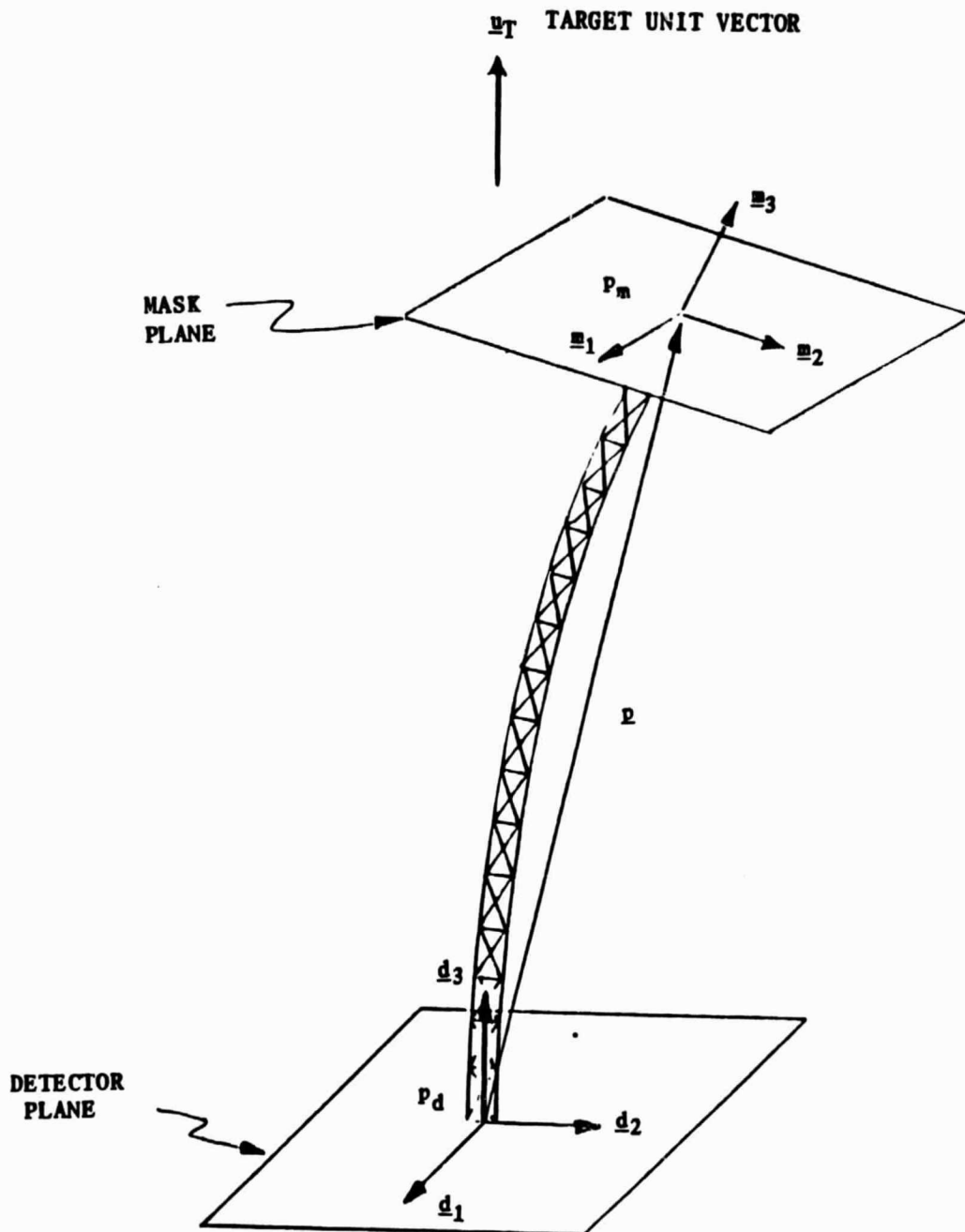
The geometry of the P/OF sensor is illustrated in Figure 5. The P/OF boom is modeled as a flexible body attached to the IPS (Body 2) in a cantilever sense, i.e. the attach point has zero elastic deformation in both rotation and translation. If we let the vector \underline{p} locate the tip of the boom (mask) wrt the attach point of the boom then the relative tilt and displacement measured by the sensor is simply the elastic deformation at node \underline{p} .

$$\underline{p} = \underline{p}^j + \underline{u}^j(\underline{p})$$

where \underline{p}^j is "rigid body location" of the mask node point, j is a body index identifier and $\underline{u}^j(\underline{p})$ is the elastic deformation at node \underline{p} .

P/OF ALIGNMENT GEOMETRY

FIGURE 5



$$\begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} = \sum_{k=1}^{NM_j} \underline{\phi}_k^j(P) \eta_k^j$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \underline{u}^j(P) = \sum_{k=1}^{NM_j} \underline{\phi}_k^j(P) \eta_k^j$$

where η_k^j are the generalized coordinates of the body j , $\underline{\phi}_k^j(P)$ and $\underline{\phi}_k^j(P)$ are the mode slopes and mode displacements respectively of body j node P .

The line of sight errors are defined in terms of the target vector \underline{u}_T and the mask location vector \underline{p} .

$$\underline{u}_p \times \underline{u}_T = \sin \alpha \underline{u}_d$$

α = line of sight angle

Let the target vector be nominally located along the bore sight axis of the P/OF.

$$\underline{u}_T = \underline{d}_3$$

$$\underline{u}_p = \sin \delta_2 \underline{d}_1 - \cos \delta_2 \sin \delta_1 \underline{d}_2 + \cos \delta_2 \cos \delta_1 \underline{d}_3$$

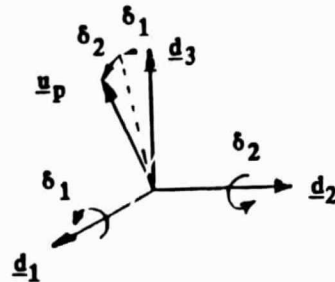
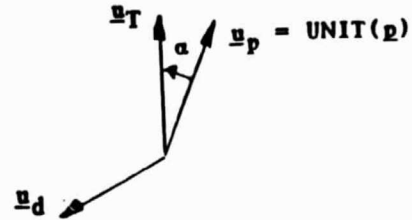
$$\underline{u}_p \approx \delta_2 \underline{d}_1 - \delta_1 \underline{d}_2 + \underline{d}_3$$

$$\underline{u}_p \times \underline{u}_T = -\delta_1 \underline{d}_1 - \delta_2 \underline{d}_2$$

If we define δ_1 and δ_2 as the line of sight errors about the \underline{d}_1 and \underline{d}_2 axes respectively then they can be computed as follows:

$$\begin{bmatrix} \delta_1 \\ \delta_2 \\ 0 \end{bmatrix} = -\underline{u}_p \times \underline{u}_T$$

The TREEIOPS input data needed to define all the P/OF sensors are listed in Table 7.



P/OF SENSOR DEFINITION

TABLE 7

1.) IPS ACCELEROMETER (X)

TYPE = ACCELEROMETER

ID# = 1

ATTACH POINT BODY ID# = 1

NODE ID# = 3

INPUT AXIS = 1,0,0

2.) IPS ACCELEROMETER (Y)

TYPE = ACCEL

ID# = 2

ATTACH POINT BODY ID# = 1

NODE ID# = 3

INPUT AXIS = 0,1,0

3.) IPS RATE GYRO (X)

TYPE = GYRO

ID# = 3

MOUNTING POINT BODY ID# = 2

NODE ID# = 3

INPUT AXIS = 1,0,0

4.) IPS RATE GYRO (Y)

TYPE = GYRO

ID# = 4

MOUNTING POINT BODY ID# = 2

NODE ID# = 3

INPUT AXIS = 0,1,0

TABLE 7
(CONTINUED)

5.) IPS RATE GYRO (Z)

TYPE = GYRO

ID# = 5

MOUNTING POINT BODY ID# = 2

NODE ID# = 3

INPUT AXIS = 0,0,1

6.) P/OF LINE OF SIGHT SENSOR

TYPE = LOS

ID# = 99

MOUNTING POINT BODY ID# = 3

NODE ID# = 2

TARGET UNIT VECTOR = 0,0,1

4.1.3 Actuator Math Models

The P/OF simulation has a total of 9 actuators; 3 torque motors mounted on the gimbal axes of the IPS and 6 reaction jets for the shuttle orbiter vernier RCS control system. Each of these are standard TREETOPS actuators and the input data necessary for their definition are presented in Table 8.

P/OF ACTUATOR DEFINITION

TABLE 8

1.) IPS TORQUE MOTOR - ROLL - X

TYPE = TORQUE MOTOR

ID# = 1

MOUNTING POINT HINGE # = 2

AXIS # = 1

2.) IPS TORQUE MOTOR - PITCH - Y

TYPE = TORQUE MOTOR

ID# = 2

MOUNTING POINT HINGE # = 2

AXIS # = 2

3.) IPS TORQUE MOTOR - YAW - Z

TYPE = TORQUE MOTOR

ID# = 3

MOUNTING POINT HINGE # = 2

AXIS # = 3

TABLE 8
(CONTINUED)

4,5,6,7,8,9.) VERNIER RCS JETS

| <u>TYPE</u> | <u>S.#</u> | <u>MOUNTING POINT</u> | | <u>OUTPUT AXIS</u> |
|-------------|------------|-----------------------|---------------|--------------------------|
| | | <u>BODY #</u> | <u>NODE #</u> | |
| JET | 4 | 1 | 4 | .03265, -.69625, .71706 |
| JET | 5 | 1 | 5 | .03265, .69625, .71706 |
| JET | 6 | 1 | 6 | 0, -.99967, .02582 |
| JET | 7 | 1 | 7 | 0, .99967, .02582 |
| JET | 8 | 1 | 8 | -.02656, -.28662, .95768 |
| JET | 9 | 1 | 9 | -.02656, .28662, .95768 |

4.1.4 Controller Math Model

TREETOPS has the capability for simulating three types of controllers; continuous, discrete and user defined. The P/OF simulation has 3 separate and distinct control systems and they are all embedded within the user defined controller. An overview of the user controller is presented in the interconnect diagram, Figure 6, which also defines the inputs and outputs. The 3 separate control systems are 1) the roll gimbal pointing control, 2) the pitch gimbal pointing control and 3) the vernier RCS orbiter attitude control.

ROLL AND PITCH GIMBAL POINTING CONTROL

The roll and pitch gimbal controller both utilize the full state controller discussed in Section 4.2. The full state controller was developed for the pitch axis and when used for the roll axis it requires a phase change (sign reversal) on the tip deflection input, x_2 . Both pointing controllers have provisions to use the accelerometer feedbacks for disturbance rejection but the gain on these feedbacks is currently set to zero.

ORBITER VERNIER RCS CONTROL LAW

The vernier RCS control system fires jets to hold the orbiter at the initial attitude or maneuver to a commanded attitude. The inputs are attitude commands supplied by the user via user controller inputs 11, 12 and 13.

USER(11) = roll attitude command (deg)
USER(12) = pitch attitude command (deg)
USER(13) = yaw attitude command (deg)

The vernier RCS control law consists of 4 primary parts as illustrated in Figure 7.

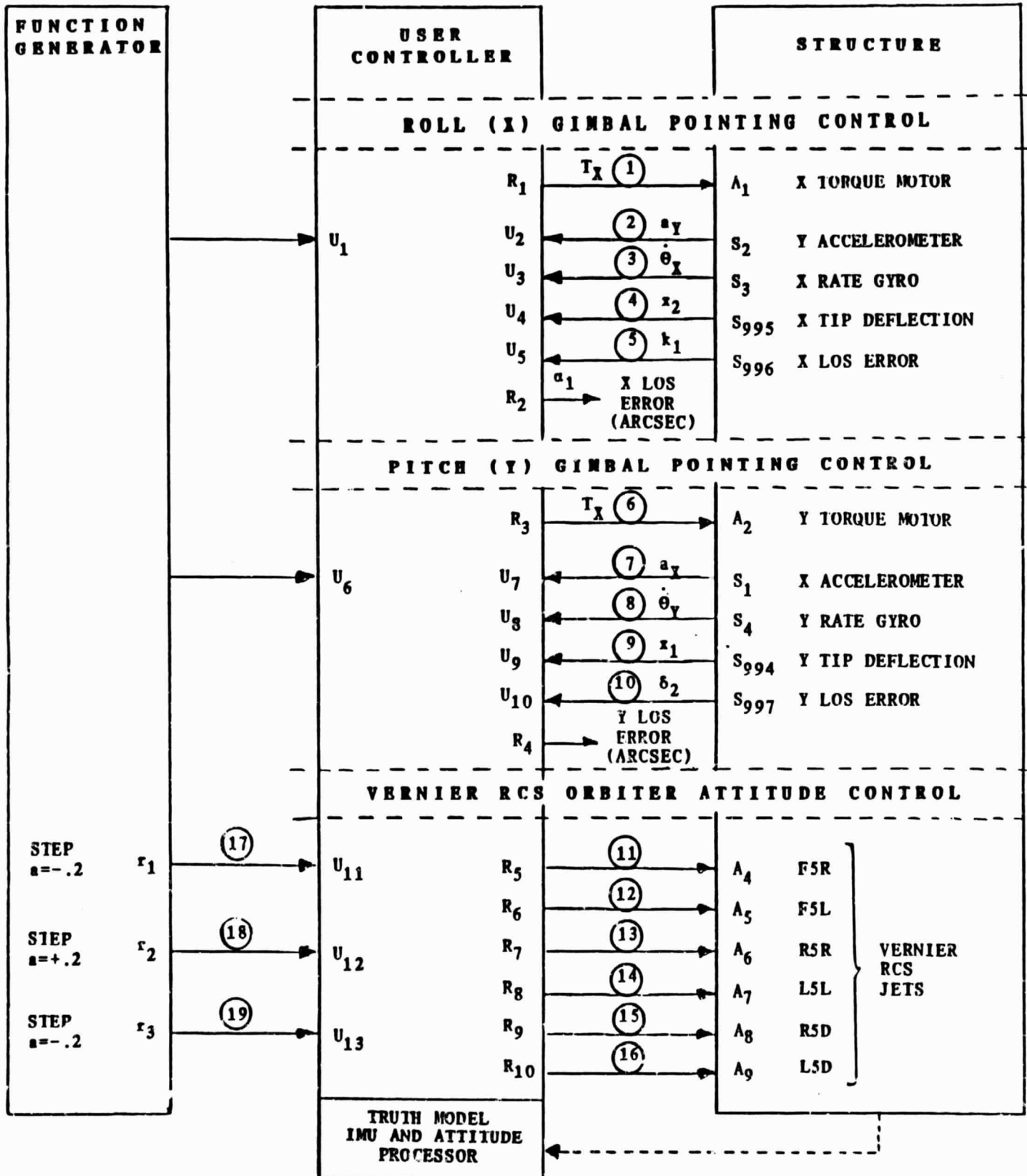
PHASE PLANE The phase plane operates on each axis (roll, pitch and yaw) to determine whether positive, negative or zero RCS control torque is required to null attitude and rate errors. Hysteresis is included to reduce jet on/off cycling. Disturbance acceleration logic sets up one sided limit cycles that remain within the attitude error deadbands while maximizing the time between jet pulses. The phase plane logic is illustrated in Figure 8.

JET SELECT The jet select logic will select a set of 1, 2 or 3 vernier RCS jets to produce the control torque requested by the phase plane. The jets to be used are selected via a "dot product" approach. That is, the control torque produced by each jet is projected (dot product) onto the desired torque vector. The jet with the largest projection is turned on. If a second jet with at least half the projection of the maximum jet is found, then it is turned on. Similarly, if a third jet is found with 40% of the maximum projection it also is turned on.

STATE ESTIMATOR The state estimator is patterned after a fixed gain Kalman Filter. The state is extrapolated forward to each control instant (12.5 Hz) assuming that acceleration is constant over the interval. The states are then updated based on attitude measurements at 6.25 Hz rate. The estimator has three separate filters for attitude, rate and disturbance accel-

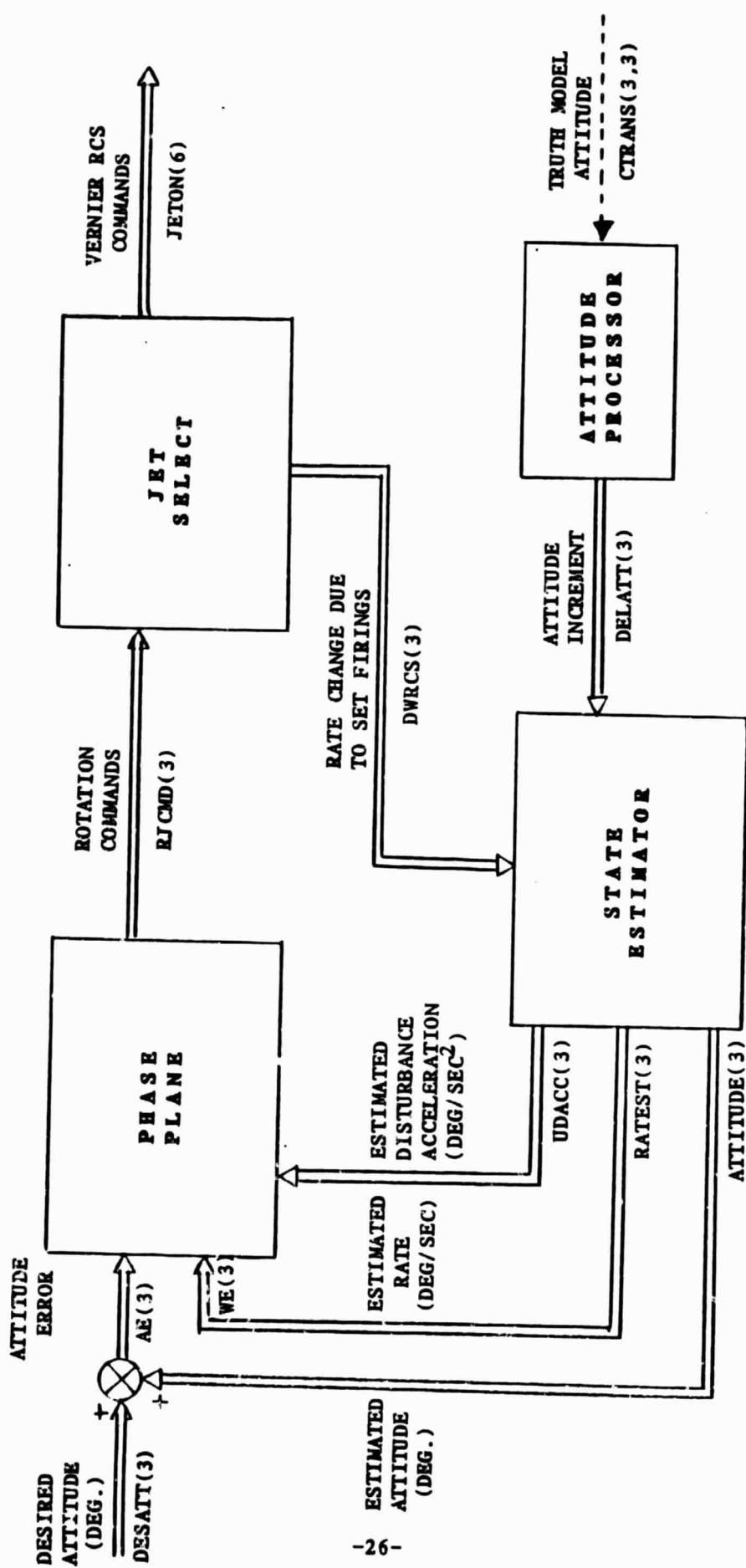
TREETOPS P/OF INTERCONNECT DIAGRAM

FIGURE 6



VERNIER RCS ATTITUDE CONTROL LAW

FIGURE 7



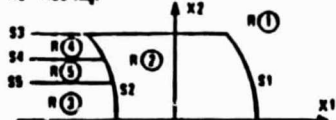
ON ORBIT PHASE PLANE LOGIC FLOW

FIGURE 8

DEFINE PHASE PLANE PARAMETERS

$$X1 = \text{SIGN}(I_{\text{REF}}) \cdot \phi_e$$

$$X2 = \text{ABS}(I_{\text{REF}})$$



DETERMINE IF IN REGION ①

$$\text{IF } X1 > S1$$

$$\text{OR}$$

$$X2 > S3$$

THEN REGION ① $RJC = \text{SIGN}(I_{\text{REF}})$

DETERMINE IF IN REGION ②

$$\text{IF } X2 \leq X1 \leq S1$$

$$\text{AND}$$

$$X2 \leq S3$$

THEN REGION ②
HYSTERESIS DISTURBANCE

DETERMINE IF IN REGION ③

$$\text{IF } X1 < S2$$

$$\text{AND}$$

$$X2 \leq S5$$

THEN REGION ③

$$RJC = \text{SIGN}(I_{\text{REF}})$$

DETERMINE IF IN REGION ④ OR REGION ⑤

IF PRIMARY SELECTED AND IF $X1 < S2$ AND $S5 < X2 < S3$

THEN $RJC = 0$

ELSE IF $X1 < S2$ AND $S4 < X2 < S3$

THEN REGION 4

IF RJC FROM PREVIOUS CYCLE $\cdot \text{SIGN}(I_{\text{REF}})$

THEN LEAVE IT AT THAT VALUE

ELSE $RJC = \text{SIGN}(I_{\text{REF}}) \cdot \text{WF RATE} \left(\frac{0.8 \text{ RL} - X2}{0.2 \text{ RL}} \right)$

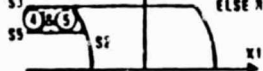
ELSE IF $X1 < S2$ AND $S5 < X2 < S4$

THEN REGION ⑤

IF RJC FROM PREVIOUS CYCLE $\cdot \text{SIGN}(I_{\text{REF}})$

THEN LEAVE IT AT THAT VALUE

ELSE $RJC = \text{SIGN}(I_{\text{REF}}) \cdot \text{WF RATE} \left(\frac{0.8 \text{ RL} - X2}{0.2 \text{ RL}} \right)$



IF IN REGION 2 USE HYSTERESIS DISTURBANCE LOGIC

REDEFINE PHASE PLANE PARAMETERS

$$Y1 = S11$$

$$Y1 = \text{SIGN}(I_{\text{REF}}) \cdot \phi_e$$

$$Y2 = \text{SIGN}(I_{\text{REF}}) \cdot \omega_{\text{MIN}}$$



DETERMINE IF IN C/R

$$\text{IF } S2 \leq Y1 < S7$$

$$\text{AND}$$

$$0 \leq Y2 \leq S3$$

THEN CS1

$$RJC = \text{SIGN}(I_{\text{REF}}) \cdot \text{WF RATE} \left(\frac{S11(Y1 - Y2)}{RL + S11(Y1)} \right)$$

SWITCHING LINE DEFINITIONS

X1, X2 LINES

$$S1 = \frac{X2^2}{2U_C} + DB$$

$$S2 = \frac{C \cdot X2^2}{2U_C} - 1.2 DB$$

$$S3 = RL$$

$$S4 = 0.8 RL \text{ IF VERNIER SELECTED}$$

$$S5 = RL - 2\omega_{\text{MIN}}$$

$$S6 = 0.8 RL \text{ IF VERNIER SELECTED}$$

$$C = 1.25 \text{ IF } |RJC| = 1$$

$$= 1.0 \text{ IF } |RJC| = 1$$

$$U_C = U \cdot \text{SIGN}(I_{\text{REF}}) \cdot U_D$$

$$U_D = \text{DISTURBANCE ACCEL}$$

$$U_C = \text{COMPUTATIONAL ACCEL}$$

$$U = \text{CONTROL ACCEL}$$

$$RL = \text{RATE LIMIT}$$

$$DB = \text{DEADBAND}$$

$$\omega_{\text{MIN}} = \text{MINIMUM IMPULSE RATE}$$

$$\text{WF RATE} = \text{SCALE FACTOR}$$

Y1, Y2 LINES

$$S1 = \frac{Y2^2}{2U_C} + DB$$

$$S2 = \frac{C \cdot Y2^2}{2U_C} - 1.2 DB$$

$$S7 = K \cdot \frac{Y2^2}{2U_C} + DB \left[\begin{array}{l} K = 1.1 \text{ IF } Y2 \geq 0 \\ K = 1.1 \text{ IF } Y2 < 0 \end{array} \right]$$

$$S8 = -RL$$

$$S10 = \frac{C \cdot Y2^2}{2U_C} + 1.2 DB$$

$$S11 = 0 \text{ IF } -1.2 DB \leq Y1 < -0.5 DB$$

$$S11 = -\sqrt{2U_D|Y1 + 0.5 DB|} \cdot \omega_{\text{MIN}}$$

$$\text{IF } -0.5 DB \leq Y1 \leq S10$$

$$S11 = 0 \text{ IF } -\sqrt{2U_D|Y1 + 0.5 DB|} \cdot \omega_{\text{MIN}} > 0$$

$$S11 = -RL + \omega_{\text{MIN}}$$

$$\text{IF } -\sqrt{2U_D|Y1 + 0.5 DB|} \cdot \omega_{\text{MIN}} < -RL$$

$$S14 = \frac{Y2^2}{2U_C} + DB$$

DETERMINE IF IN CS2

$$\text{IF } S14 < Y1 \leq S10$$

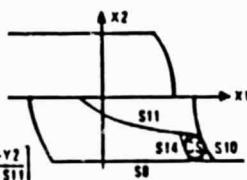
$$\text{AND}$$

$$S8 \leq Y2 < S11$$

THEN CS2

$$RJC = \text{SIGN}(I_{\text{REF}}) \cdot \text{WF RATE}$$

$$\left(\frac{S11 - Y2}{RL + S11} \right)$$



DETERMINE IF IN HS1

$$\text{IF } S7 \leq Y1 \leq S1$$

$$\text{AND}$$

$$0 \leq Y2 \leq S3$$

$$\text{OR}$$

$$S7 \leq Y1 \leq S10$$

$$\text{AND}$$

$$S11 \leq Y2 < 0$$

THEN HS1

IF RJC FROM PREVIOUS CYCLE $\cdot \text{SIGN}(I_{\text{REF}})$

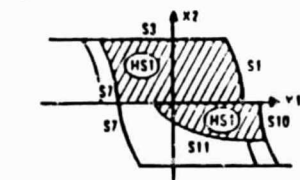
THEN LEAVE IT AT THAT VALUE

ELSE

IF FORCE FIRE = ON $RJC = \text{SIGN}(I_{\text{REF}})$

ELSE

$RJC = -\text{SIGN}(I_{\text{REF}}) \cdot \text{WF RATE} \left(\frac{Y1 - S11}{R1 - S11} \right)$



DETERMINE IF IN HS2

$$\text{IF } S7 \leq Y1 \leq S14$$

$$\text{AND}$$

$$S8 \leq Y1 < S11$$

THEN HS2

IF RJC FROM PREVIOUS CYCLE $\cdot \text{SIGN}(I_{\text{REF}})$

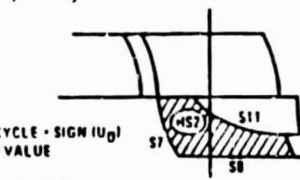
THEN LEAVE IT AT THAT VALUE

ELSE

IF FORCE FIRE = ON $RJC = \text{SIGN}(I_{\text{REF}})$

ELSE

$RJC = \text{SIGN}(I_{\text{REF}}) \cdot \text{WF RATE} \left(\frac{S11 - Y2}{RL + S11} \right)$

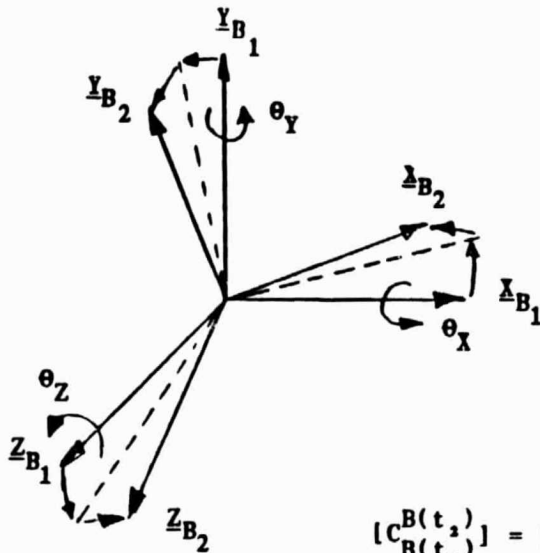


eration. The bandwidth of each filter is tuned to improve the overall performance of the system. The equations for the state estimator are presented in Figure 9.

ATTITUDE PROCESSOR The attitude processor computes the incremental attitude changes between successive sample times. It should be noted that the inputs to the phase plane are angular rate and its integral. The integral of rate is not in the strict sense an attitude measurement except for planar rotation. The integral of rate is computed by summing the incremental attitude changes where the incremental rotations are about the body axes. In the flight system the attitude is measured by the IMU. For this simulation we have neglected IMU dynamics by extracting the direction cosines directly from the truth model and computing angular increments as shown in Figure 10.

VERNIER RCS ATTITUDE PROCESSOR

FIGURE 10



$$[C_{B(t_1)}^{B(t_2)}] = [C_{B(t_2)}^I]^T [C_{B(t_1)}^I]$$

$$[C_{B(t_1)}^{B(t_2)}] \approx \begin{bmatrix} 1 & \theta_Z & -\theta_Y \\ -\theta_Z & 1 & \theta_X \\ \theta_Y & -\theta_X & 1 \end{bmatrix}$$

$$\theta_X = \left\{ [C_{B(t_1)}^{B(t_2)}] (2,3) - [C_{B(t_1)}^{B(t_2)}] (3,2) \right\} \frac{57.3}{2}$$

$$\theta_Y = \left\{ [C_{B(t_1)}^{B(t_2)}] (3,1) - [C_{B(t_1)}^{B(t_2)}] (1,3) \right\} \frac{57.3}{2}$$

$$\theta_Z = \left\{ [C_{B(t_1)}^{B(t_2)}] (1,2) - [C_{B(t_1)}^{B(t_2)}] (2,1) \right\} \frac{57.3}{2}$$

4.2 FULL STATE CONTROLLER

4.2.1 Introduction

The "full state controller" is a pointing control system for the P/OF. It uses identical control algorithms for pointing of both the roll and pitch gimbals of the instrument pointing system (IPS). The third gimbal allows rotation about the line-of-sight axis and is currently uncontrolled.

The full state controller was developed primarily for the purpose of demonstrating the closed loop performance capability of the TREETOPS simulation. As such, the control design does not represent a finished product, but it does demonstrate the feasibility of the approach and it also provides a basis for further refinement and development.

4.2.2 Salient Features

- 1.) **Uncoupled Axes** - The P/OF boom is constructed such that the bending modes are uncoupled between axes. This allows us to separate the three axis model into three single axis models for the purpose of control system design. Furthermore, the roll and pitch axis models are nearly identical enabling the use of identical control algorithms for the two axes.
- 2.) **Reduced Order Model** - The full state controller is based on optimal Linear Quadratic Gaussian (LQG) design techniques and requires full knowledge of the system state for implementation. Computational requirements are minimized by reducing the state vector dimension which for the P/OF is equivalent to reducing the number of bending modes. The model used for control design and implementation uses two modes per axis and is a derivative of the verification model which uses eight modes; four in the roll axis, three in pitch and the final one is a torsion mode.
- 3.) **Sensors** - Three sensors are used for each axis. The first senses line of sight error which is the basic control variable of the P/OF. The second measures bending deformation of the boom tip with respect to its base allowing the control system to null bending motion. The third sensor is a rate gyro mounted on the IPS and used to provide inner loop stability. The TREETOPS simulation uses these sensors together with an observer to estimate the state vector rather than simply extracting the state from the simulation for use in the controller.

4.2.3 Design Methodology

The full state controller is based on LQG design techniques and was accomplished using two computer programs; TREETOPS and HONEY-X. The basic procedure is to first create a multi-body, 3-axis, non-linear model on TREETOPS. Second, use the TREETOPS linearization option to compute a linear model for input to HONEY-X. Third, develop a single axis model and design the control system using HONEY-X. Fourth, install two copies (one each for roll and pitch) of the single axis controller into TREETOPS for evaluation in a

fully coupled, non-linear simulation.

The TREETOPS model used for the design process has 13 degrees of freedom; 3 rotation and 3 translation DOF's for the orbiter with respect to inertial space, 3 rotation DOF's for the IPS gimbals and 4 bending DOF's for the P/OF boom. Each DOF is described by a second order differential equation which means the state vector dimension will be 26.

The linear model obtained from TREETOPS has the following standard form:

$$\dot{\underline{x}} = \underline{A}\underline{x} + \underline{B}\underline{u}$$

$$\underline{r} = \underline{C}\underline{x} + \underline{D}\underline{u}$$

where:

\underline{x} = state vector dimensioned 26

\underline{u} = input vector dimensioned 9

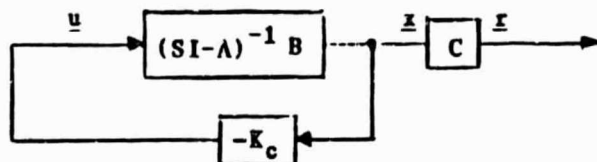
\underline{r} = output vector dimensioned 12

and the constant coefficient matrix quadruples are dimensioned A(26,26), B(26,9), C(12,26), D(12,9).

The input, output and state vectors are defined in Figure 11.

The single axis linear model was obtained from the three axis model by selecting the pitch axis variables and deleting the rest to form the model presented in Figure 12.

We can now compute an "optimal" controller for this system using LQG design techniques. This controller has the form shown below.



The control gain K_c is defined as follows:

For the cost function

$$J = 1/2 \int_0^\infty (\underline{x}^T \underline{Q} \underline{x} + \underline{u}^T \underline{R} \underline{u}) dt$$

the optimal control is given by

$$\underline{K}_c = \underline{R}^{-1} \underline{B}^T \hat{\underline{K}}$$

where $\hat{\underline{K}}$ is a solution of the non-linear matrix equation

$$-\hat{\underline{K}}\underline{A} - \underline{A}^T \hat{\underline{K}} + \hat{\underline{K}}\underline{B}\underline{R}^{-1}\underline{B}^T \hat{\underline{K}} - \underline{Q} = 0$$

**PINHOLE/OCCULTER FACILITY STATE VECTOR DEFINITION
(REDUCED ORDER MODEL)**

FIGURE 11

STATE VECTOR

| \underline{x} | |
|-----------------|---------------------|
| 1 | $\dot{\theta}_1$ |
| 2 | $\dot{\theta}_2$ |
| 3 | $\dot{\theta}_3$ |
| 4 | \dot{y}_1 |
| 5 | \dot{y}_2 |
| 6 | \dot{y}_3 |
| 7 | $\dot{\theta}_{21}$ |
| 8 | $\dot{\theta}_{22}$ |
| 9 | $\dot{\theta}_{23}$ |
| 10 | η_3^1 |
| 11 | η_3^2 |
| 12 | η_3^3 |
| 13 | η_3^4 |
| <hr/> | |
| 14 | θ_1 |
| 15 | θ_2 |
| 16 | θ_3 |
| 17 | y_1 |
| 18 | y_2 |
| 19 | y_3 |
| 20 | θ_{21} |
| 21 | θ_{22} |
| 22 | θ_{23} |
| 23 | η_3^1 |
| 24 | η_3^2 |
| 25 | η_3^3 |
| 26 | η_3^4 |

ROTATION OF BODY B1 (ORBITER) WRT INERTIAL FRAME (rows 1-3)
 TRANSLATION OF BODY B1 (ORBITER) WRT INERTIAL FRAME (rows 4-6)
 ROTATION OF BODY B2 (IPS) WRT BODY B1 (ORBITER) (rows 7-9)
 BENDING MODES OF P/OF BOOM, BODY B3 (rows 10-13)
 GENERALIZED COORDINATES CORRESPONDING TO THE ABOVE GENERALIZED VELOCITIES (rows 14-26)

INPUT (ACTUATOR COMMANDS)

| \underline{u} | |
|-----------------|-------|
| 1 | T_X |
| 2 | T_Y |
| 3 | T_Z |
| 4 | J_1 |
| 5 | J_2 |
| 6 | J_3 |
| 7 | J_4 |
| 8 | J_5 |
| 9 | J_6 |

IPS TORQUE MOTOR COMMANDS (rows 1-3)
 ORBITER VRCS JET COMMANDS (rows 4-9)

OUTPUT (SENSORS)

| \underline{z} | |
|-----------------|------------------|
| 1 | a_X |
| 2 | a_Y |
| 3 | $\dot{\theta}_X$ |
| 4 | $\dot{\theta}_Y$ |
| 5 | $\dot{\theta}_Z$ |
| 6 | β_X |
| 7 | β_Y |
| 8 | β_Z |
| 9 | x_1 |
| 10 | x_2 |
| 11 | δ_1 |
| 12 | δ_2 |

IPS ACCELEROMETERS (rows 1-2)
 IPS RATE GYROS (rows 3-5)
 P/OF MASK TILT (rows 6-8)
 P/OF BOOM TIP DEFLECTION (rows 9-10)
 LINE-OF-SIGHT ANGLE (rows 11-12)

PINHOLE/OCCULTER FACILITY SINGLE AXIS LINEAR MODEL

FIGURE 12

$$\begin{matrix} \mathbf{x} \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{matrix} = \overbrace{\begin{matrix} \mathbf{A} \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{matrix}} + \overbrace{\begin{matrix} \mathbf{B} \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{matrix}} \mathbf{u}$$

$$\begin{bmatrix} \theta_{22} \\ \eta_3^1 \\ \eta_3^3 \\ \theta_{22} \\ \eta_3^1 \\ \eta_3^3 \end{bmatrix} = \begin{bmatrix} 0 & 5.065E-4 & 2.018E-4 & 0 & 1.387E-2 & 1.183E-1 \\ 0 & -2.125E-2 & -7.377E-3 & 0 & -5.819E-1 & -4.326 \\ 0 & -4.287E-3 & -6.035E-2 & 0 & -1.174E-1 & -35.396 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_{22} \\ \eta_3^1 \\ \eta_3^3 \\ \theta_{22} \\ \eta_3^1 \\ \eta_3^3 \end{bmatrix} + \begin{bmatrix} 8.037E-5 \\ -2.934E-3 \\ -6.791E-4 \\ 0 \\ 0 \\ 0 \end{bmatrix} [T_Y]$$

$$\begin{matrix} \mathbf{x} \\ \cdot \\ \cdot \\ \cdot \end{matrix} = \overbrace{\begin{matrix} \mathbf{C} \\ \cdot \\ \cdot \\ \cdot \end{matrix}} + \overbrace{\begin{matrix} \mathbf{D} \\ \cdot \\ \cdot \\ \cdot \end{matrix}} \mathbf{u}$$

$$\begin{bmatrix} \theta_Y \\ x_1 \\ \delta_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -.02935 \\ 0 & 0 & 0 & .999983 & .03125 & -9.172E-4 \end{bmatrix} \begin{bmatrix} \theta_{22} \\ \eta_3^1 \\ \eta_3^3 \\ \theta_{22} \\ \eta_3^1 \\ \eta_3^3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} [T_Y]$$

The cost functions Q and R were chosen to penalize line of sight error δ_2 and torque motor commands T_y respectively.

$$\delta_2 = D\hat{x}$$

where D is the third row of matrix C

$$Q = D^T \sigma^2 D$$

where σ is a normalizing function chosen as 1 arcsec.

$$\sigma^2 = (1/1 \text{ arcsec})^2 = (206264 \text{ rad})^2 = 4.25 \times 10^{10}$$

R was chosen as (1 newton-meter)²

The resulting gain matrix, K_c , was obtained using HONEY-X.

$$K_c = [728866 \quad 18874.7 \quad 126.068 \quad 206152 \quad 3968.17 \quad 201.292]$$

Now that we have a controller gain, K_c , the next step of the design process is to design an observer that will estimate the state vector. The observer will have the form

$$\dot{\hat{x}} = A\hat{x} + Bu^* + K_0(\hat{r} - \hat{x})$$

$$u^* = -K_c \hat{x}$$

$$\hat{r} = C\hat{x}$$

$$\hat{r} = C\hat{x}$$

where \hat{x} , \hat{r} are the estimated state and measurement vectors, u^* is the optimal control and K_0 is the observer gain matrix. The observer gain matrix K_0 is computed by the HONEY-X program KFILT which solves the Ricatti equation

$$A^T P + P A^T + V_1 - P C^T V_2^{-1} C P = 0$$

where A, C are the system matrices, P is the error covariance matrix and V_1 , V_2 are the plant disturbance noise and measurement noise matrices.

Since we do not have good knowledge of the system noise characteristics we used the loop shaping techniques developed by the Honeywell System Research Center to choose V_1 and V_2 and thereby obtain acceptable loop frequency responses. The state noise is assumed to enter the system through the actuators or system inputs. The measurement noise is arbitrarily set to unity and p is a ratio of measurement noise to disturbance noise which is varied until acceptable performance is achieved.

$$V_2 = I$$

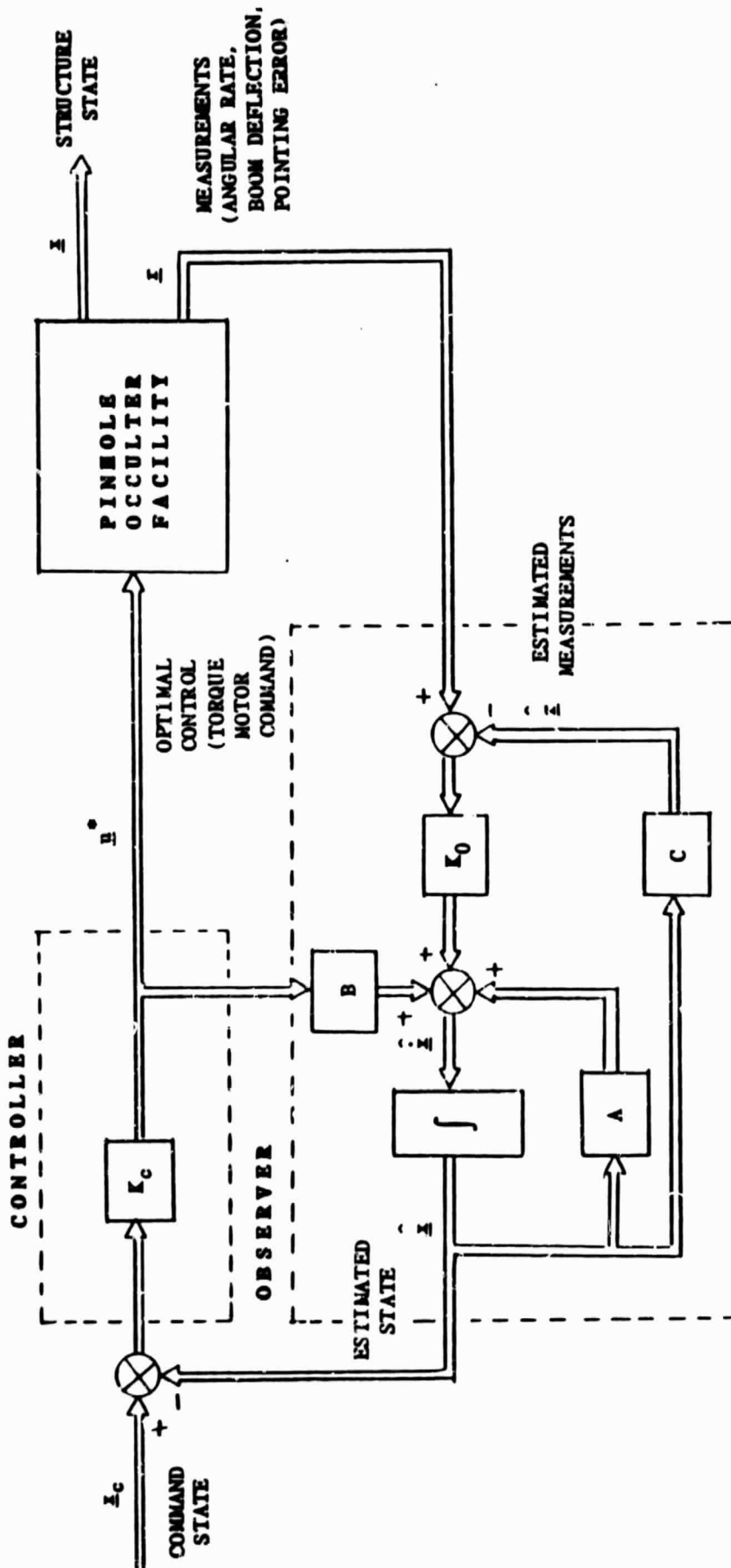
$$V_1 = \rho B B^T$$

for a value of $\rho=10^7$ the following observer gain was obtained along with

acceptable loop characteristics.

$$[K_o] = \begin{bmatrix} 2.30058E1 & -1.53860E1 & 1.91755E-1 \\ -8.26491E2 & 5.79737E2 & 9.02389E0 \\ -1.91229E2 & 1.25491E2 & 1.93675E0 \\ 6.72579E-1 & -4.77782E-1 & 8.15181E-1 \\ 1.54932E1 & 3.04520E1 & 4.70518E-1 \\ -3.65191E0 & 6.82726E0 & 9.98093E-2 \end{bmatrix}$$

The structure of the full state controller is shown in Figure 13. The values for the system matrices A, B, C are defined in Figure 12 and the controller and observer gains are defined previously in this section. For this simulation the observer was left in the continuous domain and a fourth order Runge-Kutta algorithm was used for the numerical integration. In future work we would cast the filter in the discrete domain and use transition matrices rather than numerical integration to propagate the state between control sample instants.



FULL STATE CONTROLLER

for

P/OF LINE-OF-SIGHT POINTING

FIGURE 13

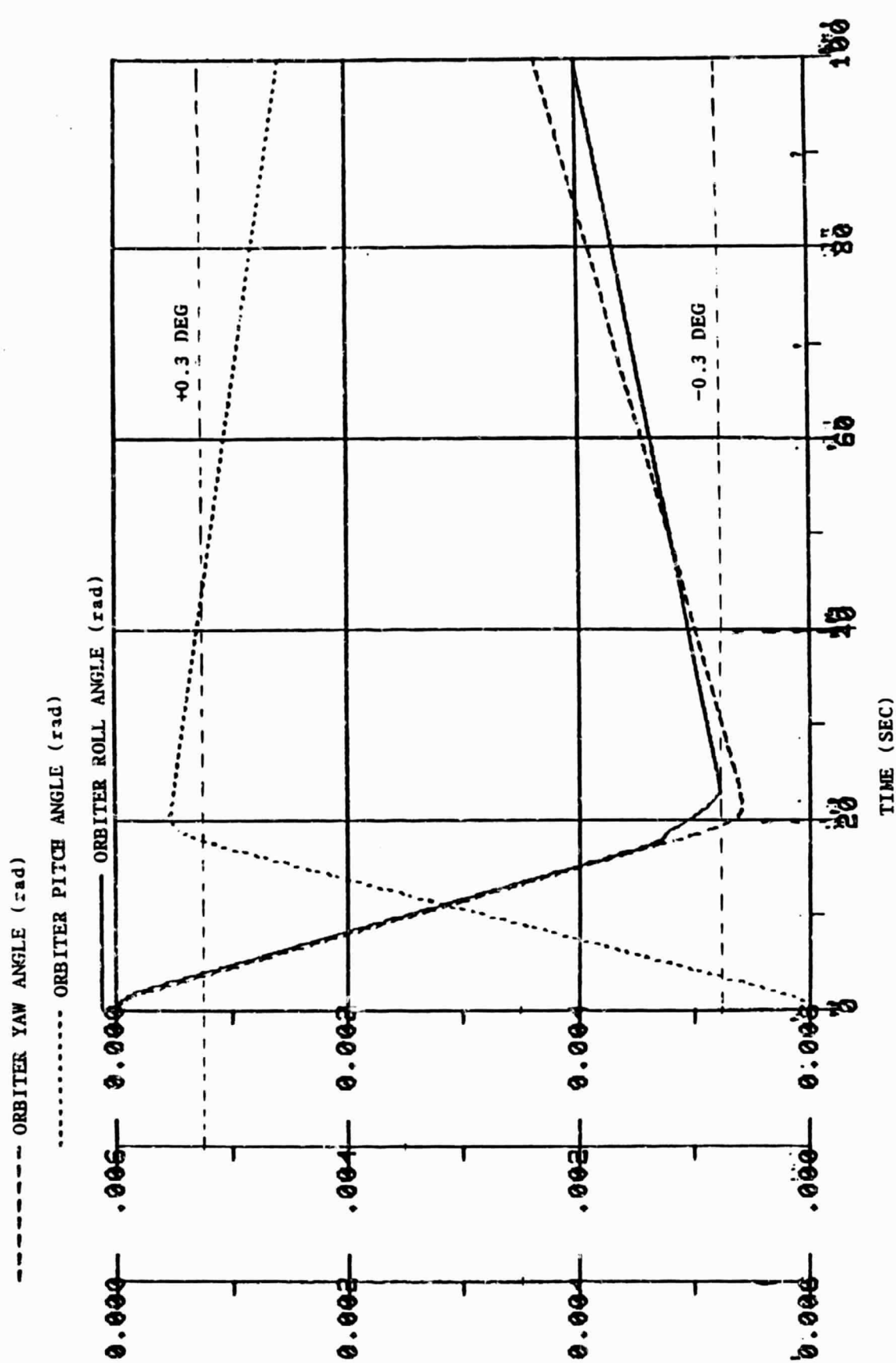
4.3 PERFORMANCE ANALYSIS

The TREETOPS simulation was used to determine the transient response of the P/OF pointing control system to disturbance inputs. We chose a three axis orbiter VRCS attitude maneuver as the disturbance. The attitude command was -0.2 , $+0.2$ and -0.2 degrees for the roll, pitch and yaw axes respectively. This combination was selected because it produces a worst case combination of jet pulses with pulse durations as long as 4.5 seconds. The orbiter attitude is shown in Figure 14 and rates in Figure 15. The attitude reached 0.3 deg in each axis before the maneuver was completed. This is equal to the command, $+0.2^\circ$, plus the attitude deadband, $+0.1^\circ$.

The P/OF pointing error corresponding to this disturbance is shown in Figure 16. During the period when the orbiter is maneuvering there is a steady state error because the system has no integral control. The transient errors are damped to less than 1 arcsec within 10 seconds. The torque motor outputs necessary to stabilize pointing are shown in Figure 17.

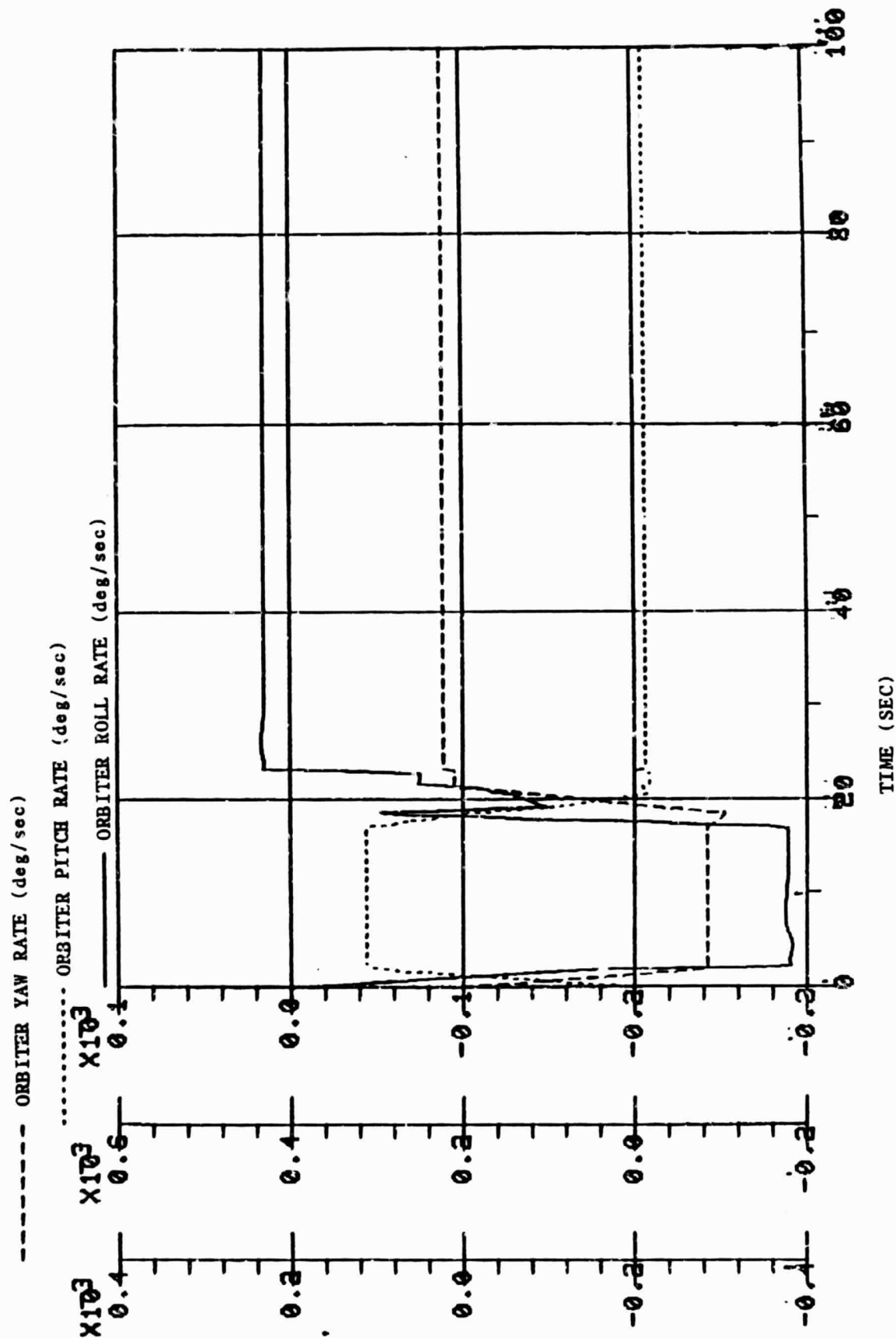
ORBITER ATTITUDE vs. TIME
for a 3 axis VRCS maneuver

FIGURE 14



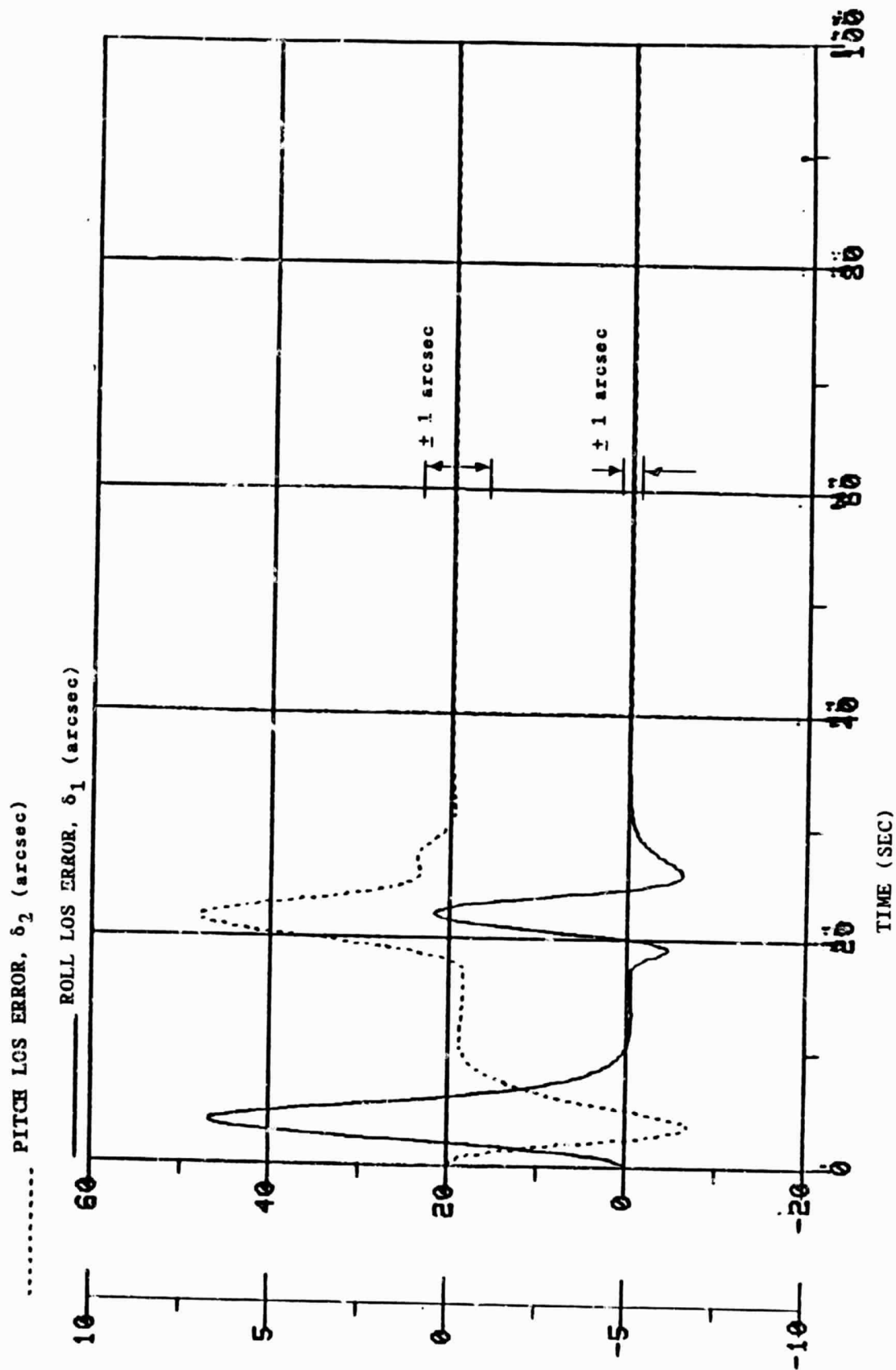
ORBITER ANGULAR RATES vs. TIME
for a 3 axis VRCS maneuver

FIGURE 15



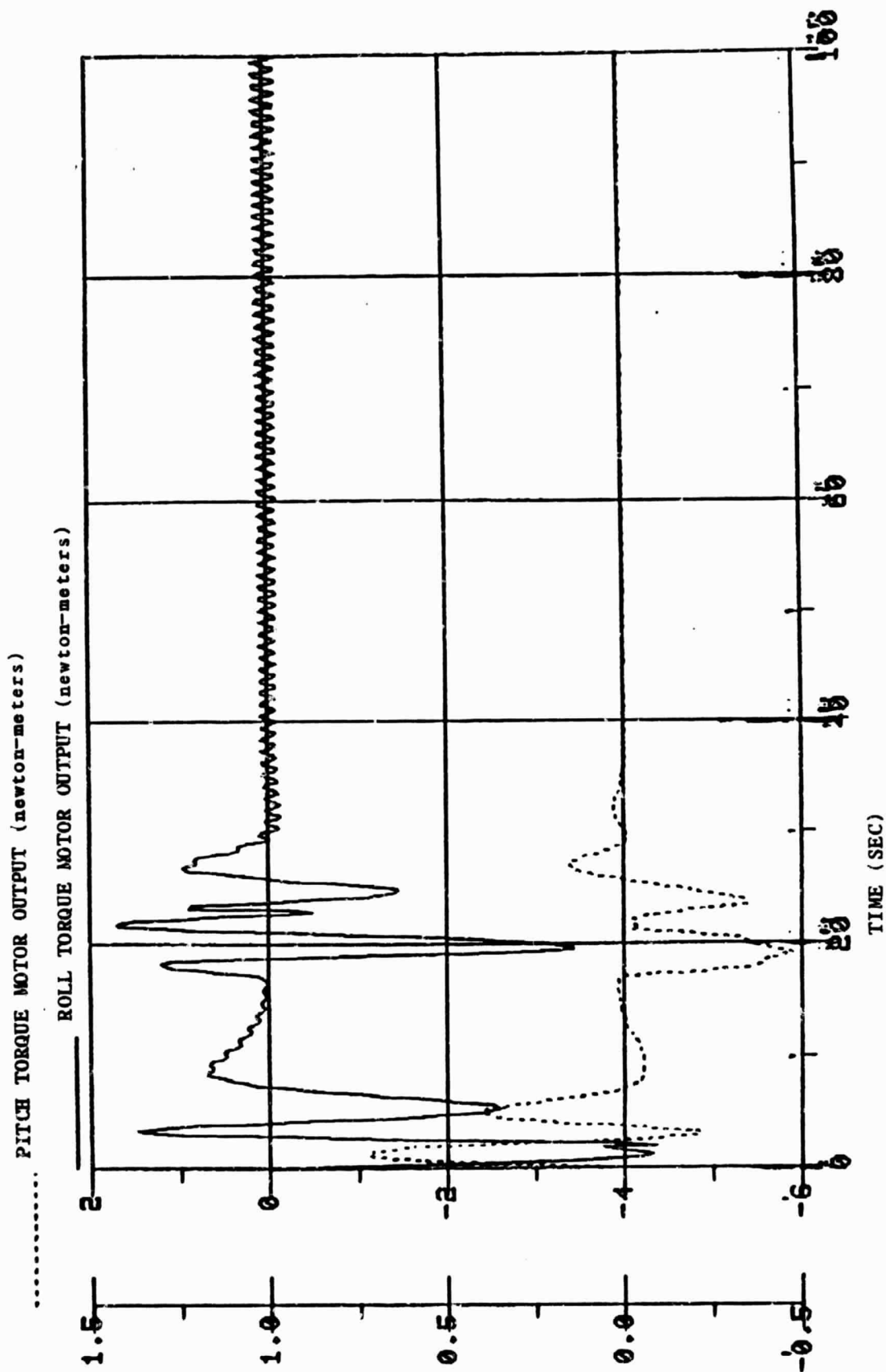
P/OF LINE-OF-SIGHT POINTING ERROR
vs. TIME during a 3 axis VRCS maneuver

FIGURE 16



P/OF TORQUE MOTOR OUTPUTS vs. TIME
for a 3 axis VRCS maneuver

FIGURE 17



5.0 COMPUTER LISTINGS

In order to completely reproduce the results presented in Section 4.3, a user needs a copy of the TREETOPS program, the TREETOPS input (INT file) for the P/OF simulation, the modal data file and the user supplied control subroutine. The TREETOPS program is documented in other sources, the input data file (RJVP6.INT;1) and user control source program (POFCONT.FOR) are included herein for the sake of completeness.

12/20/84

```

$ TY ROFCNT.FOR
SUBROUTINE UCONTROL(TIME,U,R)
DIMENSION DESATT(3),U(1),R(1),JETON(6),Y(3),X1(10),X2(10)
*,XC(10)
DATA IFIRST /0/

C
IF(IFIRST.EQ.1)GO TO 100
C INITIALIZE THE VERNIER RCS CONTROLLER
C AND THE FULL STATE GIMBAL CONTROLLER.
ACGAIN=0
INIT=-1
CALL FCS(INIT,DESATT,JETON)
INIT=0
CALL FCS(INIT,DESATT,JETON)
DT=.02
CALL FSCONT(INIT,DT,X1,XC,Y,TEMP)
C CALL FSCONT(INIT,DT,X2,XC,Y,TEMP)
I125=4
INIT=1
IFIRST=1
100 CONTINUE
C ***** ROLL GIMBAL CONTROL LAW *****
Y(1)=U(3)
Y(2)=-U(4)
Y(3)=U(5)
CALL FSCONT(INIT,DT,X1,XC,Y,TEMP)
R(1)=TEMP-ACGAIN*U(2)
R(2)=206264.0*U(5)
C
C ***** PITCH GIMBAL CONTROL LAW *****
Y(1)=U(8)
Y(2)=U(9)
Y(3)=U(10)
CALL FSCONT(INIT,DT,X2,XC,Y,TEMP)
R(3)=TEMP+ACGAIN*U(7)
R(4)=206264.0*U(10)
C
C ***** ORBITER VERNIER RCS CONTROL LAW *****
I125=I125+1
IF(I125.LT.4)GO TO 101
I125=0
DESATT(1)=-U(11)
DESATT(2)= U(12)
DESATT(3)=-U(13)
CALL FCS(INIT,DESATT,JETON)
R(5)=JETON(1)*109.
R(6)=JETON(2)*109.
R(7)=JETON(3)*106.8
R(8)=JETON(4)*106.8
R(9)=JETON(5)*64.68
R(10)=JETON(6)*64.68
101 CONTINUE
RETURN
END

```

```

SUBROUTINE FCS(INIT, DESATT, JETON)
  DIMENSION DWRC(3), RJCMD(3), UDACC(3), JETON(6)
  DIMENSION RATEST(3), ATTITUDE(3), RE(3), WE(3), DESATT(3)
C
  IF(INIT)100,200,300
100 CONTINUE
C
C SET PARAMETER DEFAULT VALUES
C
  CALL STATEST(INIT, DWRC, ATTITUDE, RATEST, UDACC)
  CALL DPHPL(INIT, RE, WE, UDACC, RJCMD)
  CALL JET SELECT(INIT, RJCMD, JETON, DWRC)
  RETURN
C
C * SET INITIAL VALUES FOR FLIGHT CONTROL MODULES
C
200 CONTINUE
  CALL STATEST(INIT, DWRC, ATTITUDE, RATEST, UDACC)
  CALL JET SELECT(INIT, RJCMD, JETON, DWRC)
  RETURN
C
C * TIME HISTORY COMPUTATIONS
C
300 CONTINUE
  CALL STATEST(INIT, DWRC, ATTITUDE, RATEST, UDACC)
  DO 3 IAX=1,3
    RE(IAx)=ATTITUDE(IAx)-DESATT(IAx)
  3 WE(IAx)=RATEST(IAx)
  CALL DPHPL(INIT, RE, WE, UDACC, RJCMD)
  CALL JET SELECT(INIT, RJCMD, JETON, DWRC)
C
  RETURN
END
SUBROUTINE ATTPROC(INIT, DELATT)
C
  DIMENSION DELATT(3), CT(3,3), C(3,3)
  INCLUDE 'DEP.FOR'
  INCLUDE 'DBB.FOR'
  IF(INIT)100,200,300
100 CONTINUE
  RETURN
200 CONTINUE
  DO 201 I=1,3
    DO 202 J=1,3
202 CT(I,J)=0.0
201 CT(I,1)=1.0
  RETURN
300 CONTINUE
  CALL MXM(CT, CTRANS(1,1,1), C, 3, 3, 3, 3)
  DELATT(1)=(C(2,3)-C(3,2))*28.6479
  DELATT(2)=(C(1,3)-C(3,1))*28.6479
  DELATT(3)=(C(1,2)-C(2,1))*28.6479
  DO 301 I=1,3
    DO 301 J=1,3

```

ORIGINAL PAGE 5
OF POOR QUALITY

```

301 CT(I,J)=CTTRANS(J,I,1)
RETURN
END
SUBROUTINE STATEST(INIT,DWRCS,ATTITUDE,OMEGA1,ALPHA2)
DIMENSION DWRCS(3),ATTITUDE(3),DELATT(3),THETAM(3),
♦THETA1(3),THETA2(3),DELY1(3),DELY2(3),
♦OMEGA1(3),OMEGA2(3),ALPHA2(3)
C
IF(INIT)100,200,300
100 CONTINUE
C
C ♦♦♦♦ SET DEFAULT VALUES ♦♦♦♦
TMEAS=.16
TDAP=.08
TDAP2=TDAP/2.
ATGAIN1=.064
ATGAIN2=1.0
RGAIN1=.0016/TMEAS
RGAIN2=.013/TMEAS
ACCGAIN=6.4E-5/(TMEAS♦2)
CALL ATTPROC(INIT,DELATT)
RETURN
200 CONTINUE
C
C ♦♦♦♦ SET INITIAL CONDITIONS ♦♦♦♦
DO 201 I=1,3
DWRCS(I)=0.
ATTITUDE(I)=0.
THETAM(I)=0.
THETA1(I)=0.
THETA2(I)=0.
OMEGA1(I)=0.
OMEGA2(I)=0.
ALPHA2(I)=0.
201 CONTINUE
I625=0.
RETURN
300 CONTINUE
C
C ♦♦♦♦ EXTRAPOLATE THE STATE OF ALL THREE FILTERS ♦♦♦♦
DO 301 I=1,3
ATTITUDE(I)=ATTITUDE(I)+TDAP♦(ALPHA2(I)+TDAP2+OMEGA1(I)+DWRCS(I))
C
THETA1(I)=THETA1(I)+TDAP♦(ALPHA2(I)+TDAP2+OMEGA1(I)+DWRCS(I))
OMEGA1(I)=OMEGA1(I)+TDAP♦ALPHA2(I)+DWRCS(I)
C
THETA2(I)=THETA2(I)+TDAP♦(ALPHA2(I)+TDAP2+OMEGA2(I)+DWRCS(I))
OMEGA2(I)=OMEGA2(I)+TDAP♦ALPHA2(I)+DWRCS(I)
C
301 CONTINUE
C
C ♦♦♦♦ UPDATE THE STATES AT 6.25 HZ RATE ♦♦♦♦
I625=I625+1
IF(I625.EQ.1)GO TO 2
I625=0

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

.CALL ATTPROC(INIT,DELATT)
DO 302 I=1,3

C
  THETAM(I)=THETAM(I)+DELATT(I)

C
  ATTITUDE(I)=THETAM(I)

C
  DELY1(I)=THETAM(I)-THETA1(I)
  THETA1(I)=THETA1(I)+ATGAIN1*DELY1(I)
  OMEGA1(I)=OMEGA1(I)+RGAIN1*DELY1(I)

C
  DELY2(I)=THETAM(I)-THETA2(I)
  THETA2(I)=THETA2(I)+ATGAIN2*DELY2(I)
  OMEGA2(I)=OMEGA2(I)+RGAIN2*DELY2(I)
  ALPHA2(I)=ALPHA2(I)+ACCGAIN*DELY2(I)

C
302 CONTINUE
  2 CONTINUE
  RETURN
  END
  SUBROUTINE JET SELECT(INIT,RJCMD,JETON,DWRCS)
  DIMENSION JFN(6),RJCMD(3),JETON(6)
  DIMENSION HFAIL(6),
  *QJETON(6),DWRCS(3),DRJCMD(3),ANGINC(6,3),
  *VJCMD(3),ROTVAL(3)
  INTEGER I,J,FAILC,JETC,JETON,QJETON,
  *INIT,M,K,Q,R,S
  REAL FSFLAG,RJCMD,JFN,X,VFAILD,
  *DWRCS,DRJCMD,VJCMD,ROTVAL,
  *MREPEAT,THRESH2,THRESH3,
  *ANGINC,IJSL,KJSL,
  *VAL1,VALX,VALY,VALZ
  LOGICAL HFAIL,JFCF,VFAIL
  DATA((ANGINC(I,J),J=1,3),I=1,6)/
  *-.4401E-03 , 0.7197E-03 , -.6793E-03 ,
  *0.4401E-03 , 0.7197E-03 , 0.6793E-03 ,
  *-.7501E-03 , -.1581E-04 , 0.5561E-03 ,
  *0.7500E-03 , -.1553E-04 , -.5561E-03 ,
  *-.7800E-03 , -.3407E-03 , 0.7697E-04 ,
  *0.7800E-03 , -.3405E-03 , -.7699E-04 /

C
C *****
C   INPUTS: RJCMD(3)   = ROTATION COMMAND (1,0,-1)(R,F,Y)
C
C   OUTPUTS: JETON(6) = RCS JET COMMAND (1=ON,0= OFF)
C
C *****
  IF(INIT) 1,2,3
1 CONTINUE
  MREPEAT = 5
  THRESH2 = .50
  THRESH3 = .4
  DO 8 I = 1,6
8 QJETON(I) = 0

```



```
DO 9 I = 1,3
9 ORJCMD(I) = 0
KJSL = 0
2 RETURN
```

ORIGINAL PAGE IS
OF POOR QUALITY

THIS PROGRAM IS THE ON-ORBIT DAP JET SELECT LOGIC
FOR THE VERNIER JETS

RJCMD(3) ROTATIONAL CMDS

CONSTANTS: ANGINC = ANGULAR RATES
MREPEAT = NUMBER OF PASSES WITHOUT CMD CHANGE (5)
THRESH2 = THRESHOLD FOR 2ND VERNIER JET(.50)
THRES3 = THRESHOLD FOR VERNIER JET 3(.4)

OUTPUTS: JETON(6) = RCS JET COMMANDS
DWRCS = DELTA OMEGA RCS

```
3 CONTINUE
DO 5 I=1,6
5 JETON(I)=0
VALX = 0
VALY = 0
VALZ = 0
Q = 0
R = 0
S = 0
```

CHECK IF VERNIER CMDS ARE DIFFERENT FROM LAST PASS
OR MAX REPEAT IS EXCEEDED

```
M=0
DO 40 I = 1,3
IF((INT(RJCMD(I)).NE.ORJCMD(I)).OR.(KJSL.EQ.MREPEAT)) M=1
40 CONTINUE
IF(M.NE.1) GO TO 300
```

CHECK IF ABS(VERNIER CMD) = 1.0

IF((ABS(RJCMD(1)).EQ.1.0).OR.(ABS(RJCMD(2)).EQ.1.0).OR.
*(ABS(RJCMD(3)).EQ.1.0)) GO TO 50
GO TO 200

```
50 CONTINUE
```

CONDUCT TESTS 1-3 PER FIG. 4.2.2.2.1-20
TO SELECT VERNIER JET CMDS (JETON(1-6))

```
TEST 1 FIND MAX OF ANG INC + VECTOR(VJCMD)
DO 70 I = 1,6
VAL1 = 0
```

```

      DO 80 K = 1,3
      VAL1 = VAL1 + RJCMD(K)*ANGINC(I,K)
80  CONTINUE
      IF(VAL1.GT.VALX) GO TO 75
      GO TO 70
75  VALX = VAL1
      Q = I
70  CONTINUE
C-----
C  TEST 2 FIND 2ND MAX OF ANGINC*VECTOR(VJCMD)>THRESH2*VALX
      DO 90 I = 1,6
      VAL1 = 0
      IF(I.EQ.Q) GO TO 90
      DO 100 K = 1,3
      VAL1 = VAL1 + RJCMD(K)*ANGINC(I,K)
100  CONTINUE
      IF((VAL1.GT.(THRESH2*VALX)).AND.(VAL1.GT.VALY)) GO TO 95
      GO TO 90
95  VALY = VAL1
      R = I
90  CONTINUE
C-----
C  TEST 3 FIND 3RD MAX OF ANGINC*VECTOR(VJCMD)>THRESH3*VALX
      IF(R.EQ.0) GO TO 120
      DO 120 I = 1,6
      VAL1 = 0
      IF((I.NE.Q).AND.(I.NE.R)) GO TO 105
      GO TO 120
105  DO 110 K = 1,3
      VAL1 = VAL1 + RJCMD(K)*ANGINC(I,K)
110  CONTINUE
      IF((VAL1.GT.(THRESH3*VALX)).AND.(VAL1.GT.VALZ)) GO TO 115
      GO TO 120
115  VALZ = VAL1
      S = I
120  CONTINUE
C-----
C  SET JETON(39-44) PER RESULTS OF TESTS 1-3 (Q,R,S)
      JETON(Q)=1
      IF(R.NE.0) JETON(R) = 1
      IF(S.NE.0) JETON(S) = 1
200  CONTINUE
C  COMPUTE DELTA OMEGA RCS
      DO 220 K = 1,3
      DWRC(K)=0.0
      DO 220 I = 1,6
      IF(JETON(I).EQ.1)DWRC(K)=DWRC(K)+ANGINC(I,K)
220  CONTINUE
      KJSL = 0
      DO 260 I = 1,3
      ORJCMD(I) = INT(RJCMD(I))
260  CONTINUE
      DO 270 I = 1,6
      OJETON(I) = JETON(I)

```

270. CONTINUE

GO TO 310

300 KJSL = KJSL + 1

DO 310 I = 1,6

JETON(I) = OJETON(I)

310 CONTINUE

RETURN

END

ORIGINAL PAGE IS
OF POOR QUALITY

SUBROUTINE OPHPL(INIT,AE,WE,UDACC,RJCMD)

----- ON-ORBIT PHASE PLANE -----

DIMENSION AE(3),WE(3),DB1(3),RLIMIT(3),ACC(3),
UDACC(3),WMIN(3),RJCMD(3),BYPASS(3),WFRATE(3),
X1(3),Y1(3),X2(3),Y2(3)

IF(INIT)100,200,300

100 CONTINUE

SET DEFAULT VALUES

DO 101 I=1,3

WFRATE(I)=0.8

DB1(I)=0.1

RLIMIT(I)=.02

101 CONTINUE

ACC(1)=.01872

ACC(2)=.01096

ACC(3)=.01264

WMIN(1)=.001872

WMIN(2)=.001096

WMIN(3)=.001264

RETURN

200 CONTINUE

RETURN

300 CONTINUE

DEFINE LOCAL VARIABLES

DO 301 IAX=1,3

X1(IAx) = SIGN(1.,WE(IAx))*AE(IAx)

X2(IAx) = ABS(WE(IAx))

Y1(IAx) = SIGN(1.0,UDACC(IAx))*AE(IAx)

Y2(IAx) = SIGN(1.0,UDACC(IAx))*WE(IAx)

C = 1.0

IF(ABS(RJCMD(IAx)) .NE. 1.0) C= 1.25

DEFINE SWITCHING LINES

REV 1B DELETED ABS UDACC PER CR12710

UCP = ACC(IAx)-SIGN(1.,WE(IAx))*UDACC(IAx)

S = 0.0

SY = 0.0

```

IF(UCP.EQ. 0.0) GO TO 105
S = -(X2(IAX)*X2(IAX))/(2.0*UCP)
Y = -(Y2(IAX)*Y2(IAX))/(2.0*UCP)
105CONTINUE
C
S1 = S + DB1(IAX)
S1Y = SY + DB1(IAX)
S2 = S+C - 1.2*DB1(IAX)
S2Y = SY+C - 1.2*DB1(IAX)
S3 = RLIMIT(IAX)
S4 = 0.8*RLIMIT(IAX)
S5 = 0.6*RLIMIT(IAX)
S7 = (-1)*SIGN(1.,Y2(IAX))*(-1.)*SY-DB1(IAX)
S8 = -RLIMIT(IAX)
S10 = (-1.)*C*SY + 1.2*DB1(IAX)
IF((Y1(IAX).LT.(-.5)*DB1(IAX)).AND.(Y1(IAX).GE.(-1.2)*DB1(IAX)
))
* S11 = 0.0
IF((Y1(IAX).GE.-0.5*DB1(IAX)).AND.(Y1(IAX).LE.S10)
*) S11 = -SQRT(
* 2.0*ABS(UDACC(IAX))*((Y1(IAX)+0.5*DB1(IAX)))+WMIN(IAX)
IF(S11.GT. 0.0) S11 = 0.0
IF(S11.LT.(-RLIMIT(IAX)+WMIN(IAX)))S11=-RLIMIT(IAX)+WMIN(IAX)
S14 = -SY + DB1(IAX)
C
C PHASE PLANE CONTROL LOGIC
C
C
C REGION 1 CONTROL
C
IF((X1(IAX).GT.S1).OR.(X2(IAX).GT.S3)) GO TO 10
C
C REGION 2 CONTROL
C ♦♦DISTURBANCE HYSTERESIS REGION♦♦
C
IF((X1(IAX).GE.S2).AND.(X1(IAX).LE.S1).AND.(X2(IAX).LE.S3))
* GO TO 15
C
C REGION 3 CONTROL
C
IF((X1(IAX).LT.S2).AND.(X2(IAX).LT.S5)) GO TO 25
C
C REGION 4 CONTROL
C
IF((X1(IAX).LT.S2).AND.(S4.LE.X2(IAX)).AND.(X2(IAX).LE.S3))
* GO TO 35
C
C REGION 5 CONTROL
IF((X1(IAX).LT.S2).AND.(S5.LE.X2(IAX)).AND.(X2(IAX).LT.S4))
* GO TO 40
WRITE(6,1000)
1000 FORMAT('PHASE PLANE ERROR - FALLS THRU REGION 5 LOGIC')
GO TO 80
C
C

```

C PHASE PLANE ACTION

C REGION 1 ACTION

C 10 CONTINUE

C RJCMD(IAX) = -SIGN(1.,WE(IAX))

C GO TO 80

ORIGINAL PAGE IS
OF POOR QUALITY

C REGION 2 ACTION

C ♦♦DISTURBANCE HYSTERESIS REGION LOGIC

C REGION 2 ACTION, CS REGION CONTROL

C 15 CONTINUE

C IF(((S2Y.LE.Y1(IAX)).AND.(Y1(IAX).LT.S7).AND.(Y2(IAX).GE.0.0).AND.

C ♦ (Y2(IAX).LE.S3)).OR.((S14.LT.Y1(IAX)).AND.(Y1(IAX).LE.S10)

C ♦ .AND.(S8.LE.Y2(IAX)).AND.(Y2(IAX).LT.S11))) GO TO 20

C DISTURBANCE HYSTERESIS REGION 1

C IF(((S7.LE.Y1(IAX)).AND.(Y1(IAX).LE.S1Y).AND.(Y2(IAX).GE.0.0).AND.

C ♦ (Y2(IAX).LE.S3)).OR.((S7.LE.Y1(IAX)).AND.(Y1(IAX).LE.S10).AND.

C ♦ (S11.LE.Y2(IAX)).AND.(Y2(IAX).LT.0.0))) GO TO 21

C DISTURBANCE HYSTERESIS REGION 2

C DEFAULT DISTURBANCE HYSTERESIS

C GO TO 23

C CS REGION ACTION

C 20 CONTINUE

C RJCMD(IAX) = SIGN(1.0,UDACC(IAX))♦WFRATE(IAX)♦

C ♦((S11-Y2(IAX))/(RLIMIT(IAX)+S11))

C GO TO 80

C HYSTERESIS REGION 1 ACTION

C 21 CONTINUE

C IF(RJCMD(IAX).EQ.-SIGN(1.0,UDACC(IAX))) GO TO 80

C 22 CONTINUE

C RJCMD(IAX) = (-1.)♦SIGN(1.0,UDACC(IAX))♦WFRATE(IAX)♦

C ♦((Y2(IAX)-S11)/(RLIMIT(IAX)-S11))

C GO TO 80

C HYSTERESIS REGION 2 ACTION

C DEFAULT ACTION

```

23 CONTINUE
IF(RJCMD(IAX).EQ.SIGN(1.0,UDACC(IAX))) GO TO 80
RJCMD(IAX) = SIGN(1.0,UDACC(IAX))*WFRATE(IAX)*
* ((S11-Y2(IAX))/(RLIMIT(IAX)+S11))
GO TO 80

REGION 3 ACTION

25 CONTINUE
RJCMD(IAX) = SIGN(1.,WE(IAX))
GO TO 80

REGION 4 ACTION

35 IF(RJCMD(IAX).EQ.-SIGN(1.0,WE(IAX))) GO TO 80
RJCMD(IAX)=-SIGN(1.0,WE(IAX))*WFRATE(IAX)*((0.8*RLIMIT(IAX)
*-X2(IAX))/(0.2*RLIMIT(IAX)))
GO TO 80
40 IF(RJCMD(IAX).EQ.SIGN(1.0,WE(IAX))) GO TO 80
RJCMD(IAX)=SIGN(1.0,WE(IAX))*WFRATE(IAX)*
*((0.8*RLIMIT(IAX)-X2(IAX))/(0.2*RLIMIT(IAX)))
GO TO 80

80 CONTINUE
81 CONTINUE
301 CONTINUE
RETURN

END
SUBROUTINE FSCONT(INIT,DT,X,XC,Y,R)
  DIMENSION A(6,6),B(6),KC(6),K0(6,3)
  DIMENSION C(3,6),BK(6,6),SUM(6,6),KOC(6,6),Y(3)
  DIMENSION KOY(6),SUMX(6),BKXC(6),XD(6),X(6),ERR(6)
  DIMENSION XD0(6),XC(6)
  REAL KC,K0,KOC,KOY
  DIMENSION PHI(6),X0(6)
  IF(INIT.GT.0)GO TO 100
  DT2=DT/2.0
  DT6=DT/6.0
  -----ZERO MATRICES A,B,K0,C,KC,C AND VECTORS X,XC,XD0-----
  DO 1 I=1,6
    B(I)=0.0
    KC(I)=0.0
    X(I)=0.0
    XC(I)=0.0
    XD0(I)=0.0
  DO 1 K=1,6
    A(I,K)=0.0
1 CONTINUE
  DO 2 I=1,6
    DO 2 J=1,3
      K0(I,J)=0.0
      C(J,I)=0.0

```

2. CONTINUE

C — ENTER NON-ZERO VALUES INTO A

A(1,2)= 5.06513E-04
A(1,3)= 2.01764E-04
A(1,5)= 1.38693E-02
A(1,6)= 1.18327E-01

ORIGINAL PAGE IS
OF POOR QUALITY

A(2,2)=-2.12523E-02
A(2,3)=-7.37679E-03
A(2,5)=-5.81931E-01
A(2,6)=-4.32620E+00

A(3,2)=-4.28686E-03
A(3,3)=-6.03547E-02
A(3,5)=-1.17383E-01
A(3,6)=-3.53957E+01

A(4,1)=1.0
A(5,2)=1.0
A(6,3)=1.0

C — ENTER THE NON-ZERO VALUES OF B

B(1)= 8.03740E-05
B(2)=-2.93356E-03
B(3)=-6.79078E-04

C — ENTER NON-ZERO VALUES INTO KC

KC(1)= 7.28866E+05
KC(2)= 1.88747E+04
KC(3)= 1.26068E+02
KC(4)= 2.06152E+05
KC(5)= 3.96817E+03
KC(6)= 2.01292E+02

C — ENTER NON-ZERO VALUES INTO K0

K0(1,1)= 2.30058E+01
K0(1,2)=-1.53860E+01
K0(1,3)= 1.51755E-01

K0(2,1)=-8.26491E+02
K0(2,2)= 5.79737E+02
K0(2,3)= 9.02389E+00

K0(3,1)=-1.91229E+02
K0(3,2)= 1.25491E+02
K0(3,3)= 1.93675E+00

K0(4,1)= 6.72579E-01
K0(4,2)=-4.77782E-01
K0(4,3)= 8.15181E-01

K0(5,1)=-1.54932E+01
K0(5,2)= 3.04520E+01

```

C      K0(5,3)= 4.70518E-01
C
C      K0(6,1)=-3.65191E+00
C      K0(6,2)= 6.82726E+00
C      K0(6,3)= 9.98093E-02
C
C-----ENTER NON-ZERO C
C      C(1,1)=1.0
C      C(2,5)=1.0
C      C(2,6)=-2.93514E-02
C      C(3,4)= 9.99983E-01
C      C(3,5)= 3.12500E-02
C      C(3,6)=-9.17230E-04
C-----CALCULATE BKXC,KOC,SUM-----
C      CALL MXM(B,KC,BKC,6,1,6,1)
C      CALL MXM(K0,C,KOC,6,3,6,3)
C      DO 3 I=1,6
C      DO 3 J=1,6
3      SUM(I,J)=A(I,J)-KOC(I,J)-BKC(I,J)
C
C-----START DYNAMIC LOOP-----
C
C      100 CONTINUE
C-----COMPUTE STATE RATES,XD-----
C      CALL MXM(K0,Y,KOY,6,3,1,3)
C      CALL MXM(SUM,X,SUMX,6,6,1,6)
C      CALL MXM(BKC,XC,BKCXC,6,6,1,6)
C      DO 11 I=1,6
11      XD(I)=KOY(I)+SUMX(I)+BKCXC(I)
C-----INTEGRATE STATE VECTOR-----
C      DO 20 I=1,6
C      X0(I)=X(I)
C      PHI(I)=XD(I)
20      X(I)=X0(I)+XD(I)*DT2
C
C      CALL MXM(SUM,X,SUMX,6,6,1,6)
C      DO 21 I=1,6
C      XD(I)=KOY(I)+SUMX(I)+BKCXC(I)
C      PHI(I)=PHI(I)+2.0*XD(I)
21      X(I)=X0(I)+XD(I)*DT2
C
C      CALL MXM(SUM,X,SUMX,6,6,1,6)
C      DO 22 I=1,6
C      XD(I)=KOY(I)+SUMX(I)+BKCXC(I)
C      PHI(I)=PHI(I)+XD(I)*2.0
22      X(I)=X0(I)+XD(I)*DT
C
C      CALL MXM(SUM,X,SUMX,6,6,1,6)
C      DO 23 I=1,6
C      XD(I)=KOY(I)+SUMX(I)+BKCXC(I)
C      X(I)=X0(I)+(PHI(I)+XD(I))*DT6
23      ERR(I)=XC(I)-X(I)
C
C-----COMPUTE CONTROLLER OUTPUT,R-----
C      CALL MXM(KC,ERR,R,1,10,1,10)
C      RETURN
C-----END

```


| | | | | | | |
|---------|-------------|---|--|----------------|--------------|--------------|
| 1. 000 | SIM CONTROL | # | 0 Title header of user problem (< 40 char) | = R, J, M, G | 1. 0000 | 1. 0000 |
| 2. 000 | SIM CONTROL | # | 0 Simulation stop time(sec), Joo time(min), Core(kuds)= | = 100. 00 | 0. 0000E+00 | 100. 00 |
| 3. 000 | SIM CONTROL | # | 0 Data output delta, Initial, Final time (sec) | = 0. 20000 | 0. 20000E-01 | |
| 4. 000 | SIM CONTROL | # | 0 Integration(R=rkutta, S=sandia, U=user), Dt(sec), Fname= | = R | 0. 00000E+00 | |
| 5. 000 | SIM CONTROL | # | 0 Linearization option (N=none, L=linear), Time(sec)= | = N | 0. 00000E+00 | |
| 6. 000 | SIM CONTROL | # | 0 Restart option (N=none, R=restart), Restart fname= | = N | 0. 00000E+00 | |
| 7. 000 | SPEEDUP | # | 0 Small angle computation(All, Bypass, First, Nth) pass= | = FIRST | | |
| 8. 000 | SPEEDUP | # | 0 Mass matrix computation(All, Bypass, First, Nth) pass= | = BYPASS | | |
| 9. 000 | SPEEDUP | # | 0 Non-linear computation(All, Bypass, First, Nth) pass= | = BYPASS | | |
| 10. 000 | CONTROLLER | U | 2 Type(C=contin, D=discrete, U=user), Dt(sec), Fname= | = USER | 0. 20000E-01 | 10. 000 |
| 11. 000 | CONTROLLER | U | 2 Number of inputs, Number of outputs | = 13. 000 | | |
| 12. 000 | FUNCT GENER | # | 1 type(step, ramp, pulse, sawtooth, sine, user) | = STEP | 0. 00000E+00 | |
| 13. 000 | FUNCT GENER | # | 1 step amplitude, start time(sec) | = -0. 20000 | | |
| 14. 000 | FUNCT GENER | # | 1 >>>>***** END OF DATA *****<<<< | = | | |
| 15. 000 | FUNCT GENER | # | 2 type(step, ramp, pulse, sawtooth, sine, user) | = STEP | 0. 00000E+00 | |
| 16. 000 | FUNCT GENER | # | 2 step amplitude, start time(sec) | = 0. 20000 | | |
| 17. 000 | FUNCT GENER | # | 2 >>>>***** END OF DATA *****<<<< | = | | |
| 18. 000 | FUNCT GENER | # | 3 type(step, ramp, pulse, sawtooth, sine, user) | = STEP | 0. 00000E+00 | |
| 19. 000 | FUNCT GENER | # | 3 step amplitude, start time(sec) | = -0. 20000 | | |
| 20. 000 | FUNCT GENER | # | 3 >>>>***** END OF DATA *****<<<< | = | | |
| 21. 000 | FUNCT GENER | # | 4 type(step, ramp, pulse, sawtooth, sine, user) | = PULSE | 0. 00000E+00 | 0. 64000 |
| 22. 000 | FUNCT GENER | # | 4 puls amplitude, start time, stop time(sec) | = 106. 80 | | |
| 23. 000 | FUNCT GENER | # | 4 >>>>***** END OF DATA *****<<<< | = | | |
| 24. 000 | SENSOR | # | 1 Type(Resolver, Tach, Gyro, Wdot, Accel, Vel, Pos), ID# | = ACCEL | 1. 0000 | |
| 25. 000 | SENSOR | # | 1 ID# of body, index of node location | = 1. 0000 | | |
| 26. 000 | SENSOR | # | 1 Input axis unit vector (body coord) X, Y, Z | = 1. 0000 | 0. 00000E+00 | 0. 00000E+00 |
| 27. 000 | SENSOR | # | 1 >>>>***** END OF DATA *****<<<< | = | | |
| 28. 000 | SENSOR | # | 2 Type(Resolver, Tach, Gyro, Wdot, Accel, Vel, Pos), ID# | = ACCEL | 2. 0000 | |
| 29. 000 | SENSOR | # | 2 ID# of body, index of node location | = 1. 0000 | | |
| 30. 000 | SENSOR | # | 2 Input axis unit vector (body coord) X, Y, Z | = 0. 00000E+00 | 0. 00000E+00 | 0. 00000E+00 |
| 31. 000 | SENSOR | # | 2 >>>>***** END OF DATA *****<<<< | = | | |
| 32. 000 | SENSOR | # | 3 Type(Resolver, Tach, Gyro, Wdot, Accel, Vel, Pos), ID# | = GYRO | 3. 0000 | |
| 33. 000 | SENSOR | # | 3 ID# of body, index of node location | = 2. 0000 | | |
| 34. 000 | SENSOR | # | 3 Input axis unit vector (body coord) X, Y, Z | = 1. 0000 | 0. 00000E+00 | 0. 00000E+00 |
| 35. 000 | SENSOR | # | 3 >>>>***** END OF DATA *****<<<< | = | | |
| 36. 000 | SENSOR | # | 4 Type(Resolver, Tach, Gyro, Wdot, Accel, Vel, Pos), ID# | = GYRO | 4. 0000 | |
| 37. 000 | SENSOR | # | 4 ID# of body, index of node location | = 2. 0000 | | |
| 38. 000 | SENSOR | # | 4 Input axis unit vector (body coord) X, Y, Z | = 0. 00000E+00 | 0. 00000E+00 | 0. 00000E+00 |
| 39. 000 | SENSOR | # | 4 >>>>***** END OF DATA *****<<<< | = | | |
| 40. 000 | SENSOR | # | 5 Type(Resolver, Tach, Gyro, Wdot, Accel, Vel, Pos), ID# | = GYRO | 5. 0000 | |
| 41. 000 | SENSOR | # | 5 ID# of body, index of node location | = 2. 0000 | | |
| 42. 000 | SENSOR | # | 5 Input axis unit vector (body coord) X, Y, Z | = 0. 00000E+00 | 0. 00000E+00 | 1. 0000 |
| 43. 000 | SENSOR | # | 5 >>>>***** END OF DATA *****<<<< | = | | |
| 44. 000 | SENSOR | # | 99 Type Resolver, Tach, Gyro, Wdot, Accel, Vel, Pos, ID# | = LOS | 99. 000 | |
| 45. 000 | SENSOR | # | 99 ID# of body, index of node location | = 3. 0000 | | |
| 46. 000 | SENSOR | # | 99 pinhole target unit vector (inertial coord) X, Y, Z | = 0. 00000E+00 | 0. 00000E+00 | 1. 0000 |
| 47. 000 | SENSOR | # | 99 >>>>***** END OF DATA *****<<<< | = | | |
| 48. 000 | ACTUATOR | # | 1 Type (J=jet, H=hydcyl, C=cmg, T=torque, B=brake), ID# | = T | 1. 0000 | |
| 49. 000 | ACTUATOR | # | 1 ID# of hinge, index of rotation axis | = 2. 0000 | | |
| 50. 000 | ACTUATOR | # | 1 >>>>***** END OF DATA *****<<<< | = | | |
| 51. 000 | ACTUATOR | # | 2 Type (J=jet, H=hydcyl, C=cmg, T=torque, B=brake), ID# | = T | 2. 0000 | |
| 52. 000 | ACTUATOR | # | 2 ID# of hinge, index of rotation axis | = 2. 0000 | | |
| 53. 000 | ACTUATOR | # | 2 >>>>***** END OF DATA *****<<<< | = | | |
| 54. 000 | ACTUATOR | # | 3 Type (J=jet, H=hydcyl, C=cmg, T=torque, B=brake), ID# | = T | 3. 0000 | |
| 55. 000 | ACTUATOR | # | 3 ID# of hinge, index of rotation axis | = 2. 0000 | | |
| 56. 000 | ACTUATOR | # | 3 >>>>***** END OF DATA *****<<<< | = | | |
| 57. 000 | ACTUATOR | # | 4 Type (J=jet, H=hydcyl, C=cmg, T=torque, B=brake), ID# | = J | 4. 0000 | |
| 58. 000 | ACTUATOR | # | 4 ID# of body, index of node location | = 1. 0000 | | |
| 59. 000 | ACTUATOR | # | 4 Output axis unit vector (body coord) X, Y, Z | = 0. 36650E-01 | -0. 69625 | 0. 71705 |
| 60. 000 | ACTUATOR | # | 4 >>>>***** END OF DATA *****<<<< | = | | |
| 61. 000 | ACTUATOR | # | 5 Type (J=jet, H=hydcyl, C=cmg, T=torque, B=brake), ID# | = J | 5. 0000 | |

| | | | | | | | |
|---------|-------------|---|----|--|---|--------------|----------|
| 62.000 | ACTUATOR | # | 5 | 10# of body, index of node location | = | 1.0000 | 5.0000 |
| 63.000 | ACTUATOR | # | 5 | Output axis unit vector(body coord) X,Y,Z | = | 0.32650E-01 | 0.69625 |
| 64.000 | ACTUATOR | # | 5 | >>>>***** END OF DATA *****<<<< | = | | |
| 65.000 | ACTUATOR | # | 6 | Type (J=jet,H=hydraulic,C=cmg,T=torque,B=brake), ID# | = | J | 6.0000 |
| 66.000 | ACTUATOR | # | 6 | 10# of body, index of node location | = | 1.0000 | 6.0000 |
| 67.000 | ACTUATOR | # | 6 | Output axis unit vector(body coord) X,Y,Z | = | 0.00000E+00 | -0.99967 |
| 68.000 | ACTUATOR | # | 6 | >>>>***** END OF DATA *****<<<< | = | | |
| 69.000 | ACTUATOR | # | 7 | Type (J=jet,H=hydraulic,C=cmg,T=torque,B=brake), ID# | = | J | 7.0000 |
| 70.000 | ACTUATOR | # | 7 | 10# of body, index of node location | = | 1.0000 | 7.0000 |
| 71.000 | ACTUATOR | # | 7 | Output axis unit vector(body coord) X,Y,Z | = | 0.00000E+00 | 0.99967 |
| 72.000 | ACTUATOR | # | 7 | >>>>***** END OF DATA *****<<<< | = | | |
| 73.000 | ACTUATOR | # | 8 | Type (J=jet,H=hydraulic,C=cmg,T=torque,B=brake), ID# | = | J | 8.0000 |
| 74.000 | ACTUATOR | # | 8 | 10# of body, index of node location | = | 1.0000 | 8.0000 |
| 75.000 | ACTUATOR | # | 8 | Output axis unit vector(body coord) X,Y,Z | = | -0.25560E-01 | -0.28662 |
| 76.000 | ACTUATOR | # | 8 | >>>>***** END OF DATA *****<<<< | = | | |
| 77.000 | ACTUATOR | # | 9 | Type (J=jet,H=hydraulic,C=cmg,T=torque,B=brake), ID# | = | J | 9.0000 |
| 78.000 | ACTUATOR | # | 9 | 10# of body, index of node location | = | 1.0000 | 9.0000 |
| 79.000 | ACTUATOR | # | 9 | Output axis unit vector(body coord) X,Y,Z | = | -0.25560E-01 | 0.28662 |
| 80.000 | ACTUATOR | # | 9 | >>>>***** END OF DATA *****<<<< | = | | |
| 81.000 | INT CONNECT | # | 1 | Source(C=contin,D=disc,U=user,S=sen,G=gen), ID# | = | U | 1.0000 |
| 82.000 | INT CONNECT | # | 1 | Destination(C=contin,D=disc,U=user,A=actuator), ID# | = | A | 1.0000 |
| 83.000 | INT CONNECT | # | 1 | >>>>***** END OF DATA *****<<<< | = | | |
| 84.000 | INT CONNECT | # | 2 | Source(C=contin,D=disc,U=user,S=sen,G=gen), ID# | = | S | 2.0000 |
| 85.000 | INT CONNECT | # | 2 | Destination(C=contin,D=disc,U=user,A=actuator), ID# | = | U | 2.0000 |
| 86.000 | INT CONNECT | # | 2 | >>>>***** END OF DATA *****<<<< | = | | |
| 87.000 | INT CONNECT | # | 3 | Source(C=contin,D=disc,U=user,S=sen,G=gen), ID# | = | S | 3.0000 |
| 88.000 | INT CONNECT | # | 3 | Destination(C=contin,D=disc,U=user,A=actuator), ID# | = | U | 3.0000 |
| 89.000 | INT CONNECT | # | 3 | >>>>***** END OF DATA *****<<<< | = | | |
| 90.000 | INT CONNECT | # | 4 | Source(C=contin,D=disc,U=user,S=sen,G=gen), ID# | = | S | 995.00 |
| 91.000 | INT CONNECT | # | 4 | Destination(C=contin,D=disc,U=user,A=actuator), ID# | = | U | 4.0000 |
| 92.000 | INT CONNECT | # | 4 | >>>>***** END OF DATA *****<<<< | = | | |
| 93.000 | INT CONNECT | # | 5 | Source(C=contin,D=disc,U=user,S=sen,G=gen), ID# | = | S | 996.00 |
| 94.000 | INT CONNECT | # | 5 | Destination(C=contin,D=disc,U=user,A=actuator), ID# | = | U | 5.0000 |
| 95.000 | INT CONNECT | # | 5 | >>>>***** END OF DATA *****<<<< | = | | |
| 96.000 | INT CONNECT | # | 6 | Source(C=contin,D=disc,U=user,S=sen,G=gen), ID# | = | U | 3.0000 |
| 97.000 | INT CONNECT | # | 6 | Destination(C=contin,D=disc,U=user,A=actuator), ID# | = | A | 2.0000 |
| 98.000 | INT CONNECT | # | 6 | >>>>***** END OF DATA *****<<<< | = | | |
| 99.000 | INT CONNECT | # | 7 | Source(C=contin,D=disc,U=user,S=sen,G=gen), ID# | = | S | 1.0000 |
| 100.000 | INT CONNECT | # | 7 | Destination(C=contin,D=disc,U=user,A=actuator), ID# | = | U | 7.0000 |
| 101.000 | INT CONNECT | # | 7 | >>>>***** END OF DATA *****<<<< | = | | |
| 102.000 | INT CONNECT | # | 8 | Source(C=contin,D=disc,U=user,S=sen,G=gen), ID# | = | S | 4.0000 |
| 103.000 | INT CONNECT | # | 8 | Destination(C=contin,D=disc,U=user,A=actuator), ID# | = | U | 8.0000 |
| 104.000 | INT CONNECT | # | 8 | >>>>***** END OF DATA *****<<<< | = | | |
| 105.000 | INT CONNECT | # | 9 | Source(C=contin,D=disc,U=user,S=sen,G=gen), ID# | = | S | 994.00 |
| 106.000 | INT CONNECT | # | 9 | Destination(C=contin,D=disc,U=user,A=actuator), ID# | = | U | 9.0000 |
| 107.000 | INT CONNECT | # | 9 | >>>>***** END OF DATA *****<<<< | = | | |
| 108.000 | INT CONNECT | # | 10 | Source(C=contin,D=disc,U=user,S=sen,G=gen), ID# | = | S | 997.00 |
| 109.000 | INT CONNECT | # | 10 | Destination(C=contin,D=disc,U=user,A=actuator), ID# | = | U | 10.000 |
| 110.000 | INT CONNECT | # | 10 | >>>>***** END OF DATA *****<<<< | = | | |
| 111.000 | INT CONNECT | # | 11 | Source(C=contin,D=disc,U=user,S=sen,G=gen), ID# | = | U | 5.0000 |
| 112.000 | INT CONNECT | # | 11 | Destination(C=contin,D=disc,U=user,A=actuator), ID# | = | A | 4.0000 |
| 113.000 | INT CONNECT | # | 11 | >>>>***** END OF DATA *****<<<< | = | | |
| 114.000 | INT CONNECT | # | 12 | Source(C=contin,D=disc,U=user,S=sen,G=gen), ID# | = | U | 6.0000 |
| 115.000 | INT CONNECT | # | 12 | Destination(C=contin,D=disc,U=user,A=actuator), ID# | = | A | 5.0000 |
| 116.000 | INT CONNECT | # | 12 | >>>>***** END OF DATA *****<<<< | = | | |
| 117.000 | INT CONNECT | # | 13 | Source(C=contin,D=disc,U=user,S=sen,G=gen), ID# | = | U | 7.0000 |
| 118.000 | INT CONNECT | # | 13 | Destination(C=contin,D=disc,U=user,A=actuator), ID# | = | A | 6.0000 |
| 119.000 | INT CONNECT | # | 13 | >>>>***** END OF DATA *****<<<< | = | | |
| 120.000 | INT CONNECT | # | 14 | Source(C=contin,D=disc,U=user,S=sen,G=gen), ID# | = | U | 8.0000 |
| 121.000 | INT CONNECT | # | 14 | Destination(C=contin,D=disc,U=user,A=actuator), ID# | = | A | 7.0000 |
| 122.000 | INT CONNECT | # | 14 | >>>>***** END OF DATA *****<<<< | = | | |

245.000 BODY
246.000 BODY
247.000 BODY

3 NODE 2 coordinates (m) x y z
3 00000 ***** END OF DATA *****
3 00000 ***** END OF DATA *****

= 0 00000E+00 0 00000E+00 32 000
=
=

ORIGINAL PAGE 15
OF POOR QUALITY